Hindawi Publishing Corporation
Computational Intelligence and Neuroscience
Volume 2013, Article ID 251905, 17 pages
http://dx.doi.org/10.1155/2013/251905

Review Article

An Overview of Bayesian Methods for Neural Spike Train Analysis

Zhe Chen[1,2]

[1] *Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, 43 Vassar Street, Cambridge, MA 02139, USA*
[2] *Picower Institute for Learning and Memory, Massachusetts Institute of Technology, Cambridge, MA 02139, USA*

Correspondence should be addressed to Zhe Chen; zhechen@mit.edu

Received 16 May 2013; Revised 10 September 2013; Accepted 23 September 2013

Academic Editor: Wei Wu

Neural spike train analysis is an important task in computational neuroscience which aims to understand neural mechanisms and gain insights into neural circuits. With the advancement of multielectrode recording and imaging technologies, it has become increasingly demanding to develop statistical tools for analyzing large neuronal ensemble spike activity. Here we present a tutorial overview of Bayesian methods and their representative applications in neural spike train analysis, at both single neuron and population levels. On the theoretical side, we focus on various approximate Bayesian inference techniques as applied to latent state and parameter estimation. On the application side, the topics include spike sorting, tuning curve estimation, neural encoding and decoding, deconvolution of spike trains from calcium imaging signals, and inference of neuronal functional connectivity and synchrony. Some research challenges and opportunities for neural spike train analysis are discussed.

1. Introduction

Neuronal action potentials or spikes are the basic language that neurons use to represent and transmit information. Understanding neuronal representations of spike trains remains a fundamental task in computational neuroscience [1, 2]. With the advancement of multielectrode array and imaging technologies, neuroscientists have been able to record a large population of neurons at a fine temporal and spatial resolution [3]. To extract ("read out") information from or inject/restore ("write in") signals to neural circuits [4], there are emerging needs for modeling and analyzing neural spike trains recorded directly or extracted indirectly from neural signals, as well as building closed-loop brain-machine interfaces (BMIs). Many good examples and applications can be found in the volumes of the current or other special issues [5, 6].

In recent years, cutting-edge Bayesian methods have gained increasing attention in the analysis of neural data and neural spike trains. Despite its well-established theoretic principle since the inception of Bayes' rule [7], Bayesian machinery has not been widely used in large-scaled data analysis until very recently. This was partially ascribed to two facts: first, the development of new methodologies and effective algorithms; second, the ever-increasing computing power. The major theoretic or methodological development has been reported in the field of statistics, and numerous algorithms were developed in applied statistics and machine learning for successful real-world applications [8]. It is time to push this research frontier to neural data analysis. With this purpose in mind, this paper provides a tutorial review on the basic theory and the state-of-the-art Bayesian methods for neural spike train analysis.

The rest of the paper is organized as follows. Section 2 presents the background information about statistical inference and estimation, Bayes' theory, and statistical characterization of neural spike trains. Section 3 reviews several important Bayesian modeling and inference methods in light of different approximation techniques. Section 4 reviews a few representative applications of Bayesian methods for neural spike train analysis. Finally, Section 5 concludes the paper with discussions on a few challenging research topics in neural spike train analysis.

2. Background

2.1. Estimation and Inference: Statistic versus Dynamic. Throughout this paper, we denote by Y the observed variables, by X the hidden variables and by θ an unknown parameter vector, and by $^\top$ the transpose operator for vector or matrix. We assume that $p(Y \mid X,\theta)$ has a regular and well-defined form of the likelihood function. For neural spike train analysis, Y typically consists of time series of single or multiple spike trains. Given a fixed time interval $(0,T]$, by time discretization we have $Y = \{Y_1, Y_2, \dots, Y_K\}$ (where $K = T/\Delta$ and Δ denotes the temporal bin size). A general statistical inference problem is stated as follows: given observations Y, estimate the unknown hidden variable X with a known θ, or estimate θ alone, or jointly estimate θ and X. The unknown variables θ and X can be either static or dynamic (e.g., time-varying with a Markovian structure). We will review the approaches that tackle these scenarios in this paper.

There are two fundamental approaches to solve the inference problem: likelihood approach and Bayesian approach. The likelihood approach [9] computes a point estimate by maximizing the likelihood function and represents the uncertainty of estimate via confidence intervals. The maximum likelihood estimate (m.l.e.) is asymptotically consistent, normal, and efficient, and it is invariant to reparameterization (i.e., functional invariance). However, the m.l.e. is known to suffer from overfitting, and therefore model selection is required in statistical data analysis. In contrast, the Bayesian philosophy lets data speak for themselves and models the unknowns (parameters, latent variables, and missing data) and uncertainties (which are not necessarily random) with probabilities or probability densities. The Bayesian approach computes the full posterior of the unknowns based on the rules of probability theory; the Bayesian approach can resolve the overfitting problem in a principled way [7, 8].

2.2. Bayesian Inference. The foundation of Bayesian inference is given by Bayes' rule, which consists of two rules: *product rule* and *sum rule*. Bayes' rule provides a way to compute the conditional, joint, and marginal probabilities. Specifically, let X and Y be two continuous random variables (r.v.); the conditional probability $p(X \mid Y)$ is given by

$$p(X \mid Y) = \frac{p(X,Y)}{p(Y)} = \frac{p(Y \mid X)\, p(X)}{\int p(Y \mid X)\, p(X)\, dX}. \tag{1}$$

If $X = \{X_i\}$ is discrete, then (1) is rewritten as

$$p(X_i \mid Y) = \frac{p(X_i, Y)}{p(Y)} = \frac{p(Y \mid X_i)\, p(X_i)}{\sum_j p(Y \mid X_j)\, p(X_j)}. \tag{2}$$

In Bayesian language, $p(Y \mid X)$, $p(X)$, and $p(X \mid Y)$ are referred to as the likelihood, prior and posterior, respectively. The Bayesian machinery consists of three types of basic operations: *normalization, marginalization,* and *expectation*, all of which involve integration. And the optimization problem consists in maximizing the posterior $p(X \mid Y)$ and finding the maximum a posteriori (MAP) estimate $X_{\mathrm{MAP}} = \arg_X \max p(X \mid Y)$. Notably, except for very few scenarios (i.e., Gaussianity), most integrations are computationally intractable or costly when dealing with high-dimensional problems. Therefore, for the sake of computational tractability, various types of approximations are often assumed at different stages of the inference procedure.

More specifically, for the state and parameter estimation problem, Bayesian inference aims to infer the joint posterior of the state and the parameter using Bayes' rule

$$\begin{aligned} p(X,\theta \mid Y) &\approx p(X \mid Y)\, p(\theta \mid Y) \\ &= \frac{p(Y \mid X,\theta)\, p(X)\, p(\theta)}{p(Y)} \\ &= \frac{p(Y \mid X,\theta)\, p(X)\, p(\theta)}{\iint p(Y \mid X,\theta)\, p(X)\, p(\theta)\, dX\, d\theta}, \end{aligned} \tag{3}$$

where the first equation assumes a factorial form of the posterior for the state and the parameter (first-stage approximation) and $p(X)$ and $p(\theta)$ denote the prior distributions for the state and parameter, respectively. The denominator of (3) is a normalizing constant known as the partition function. When dealing with a prediction problem for unseen data Y^*, we compute the posterior predictive distribution

$$p(Y^* \mid Y) = \iint p(Y^* \mid Y, \theta, X)\, p(X \mid Y)\, p(\theta \mid Y)\, dX\, d\theta \tag{4}$$

or its expected mean $\widehat{Y}^* = \mathbb{E}_{p(Y^*|Y)}[Y^*] = \iiint Y^* p(Y^* \mid Y, \theta, X) p(X \mid Y) p(\theta \mid Y) dX\, d\theta\, dY^*$.

Sometimes, instead of maximizing the posterior, Bayesian inference attempts to maximize the marginal likelihood (also known as "evidence") $p(Y)$ as follows:

$$p(Y) = \iint p(Y \mid X,\theta)\, p(X)\, p(\theta)\, dX\, d\theta. \tag{5}$$

The second-stage approximation in approximate Bayesian inference deals with the integration in computing (3), (4), or (5), which will be reviewed in Section 3.

Note. Maximum likelihood inference can be viewed as a special case of Bayesian inference, in which θ is represented by a Dirac-delta function centered at the point estimate $\widehat{\theta}_{\mathrm{m.l.e.}}$; namely, $p(\theta) = \delta(\theta - \widehat{\theta}_{\mathrm{m.l.e.}})$. Nevertheless, Bayesian inference can still be embedded into likelihood inference to estimate $p(X)$ given a point estimate of θ. The $p(X)$ can either have an analytic form (with finite natural parameters) or be represented by Monte Carlo samples; the latter approach may be viewed as a specific case of Monte Carlo expectation-maximization (EM) methods.

2.3. Characterization of Neural Spike Trains. Neural spike trains can be modeled as a simple (temporal) point process [10]. For a single neural spike train observed in $(0,T]$, we often discretize it with a small bin size Δ such that each bin contains no more than one spike. The conditional intensity

Table 1: Probability distributions for modeling neuronal spike count observations.

Distribution	Mean statistic	Variance	Note for observations Y
Binomial (p)	$\mathbb{E}[Y] = p$	$p(1-p)$	$Y \in \{0, 1\}$
Poisson (λ)	$\mathbb{E}[Y] = \lambda$	λ	$Y \geq 0, Y \in \mathbb{Z}^{+}$
Negative binomial (r, p)	$\mathbb{E}[Y] = pr/(1-p)$	$pr/(1-p)^2$	$Y \geq 0, Y \in \mathbb{Z}^{+}$ (overdispersed Poisson)
Skellam (μ_1, μ_2)	$\mathbb{E}[Y] = \mu_1 - \mu_2$	$\mu_1 + \mu_2$	$Y \in \mathbb{Z}$ (difference between two Poissons)

function (CIF), denoted as $\lambda(t \mid H_t)$, is used to characterize the spiking probability of a neural point process as follows:

$$\lambda(t \mid H_t) = \lim_{\Delta \to 0} \frac{\Pr\{\text{spike in } (t, t+\Delta] \mid H_t\}}{\Delta}, \tag{6}$$

where H_t denotes all history information available up to time t (that may include spike history, stimulus covariate, etc.). When $\lambda(t \mid H_t)$ is history independent, then the stochastic process is an inhomogeneous Poisson process. For notation simplicity, we sometimes use λ_t to replace $\lambda(t \mid H_t)$ when no confusion occurs. When Δ is sufficiently small, the product $\lambda(t \mid H_t)\Delta$ is approximately equal to the probability of observing a spike within the interval $((t-1)\Delta, t\Delta]$. Assuming that the CIF λ_t is characterized by a parameter θ and an observed or latent variable X, then the point process likelihood function is given as [11–13]

$$\begin{aligned} p(Y \mid X, \theta) = \exp\Bigg\{ & \int_0^T \log\lambda(\tau \mid \theta, X)\, dy(\tau) \\ & - \int_0^T \lambda(\tau \mid \theta, X)\, d\tau \Bigg\}, \end{aligned} \tag{7}$$

where $dy(t)$ is an indicator function of the spike presence within the interval $((t-1)\Delta, t\Delta]$. In the presence of multiple spike trains from C neurons, assuming that multivariate point process observations are conditionally independent at any time t given a new parameter θ, one then has

$$\begin{aligned} p(\mathrm{Y}_{1:C} \mid X, \theta) &= \prod_{c=1}^{C} p(Y_c \mid X, \theta) \\ &= \prod_{c=1}^{C} \exp\Bigg\{ \int_0^T \log\lambda_c(\tau \mid \theta, X)\, dy_c(\tau) \\ &\qquad - \int_0^T \lambda_c(\tau \mid \theta, X)\, d\tau \Bigg\}. \end{aligned} \tag{8}$$

Since neural spike trains are fully characterized by the CIF, the modeling goal is then turned to model the CIF, which can have a parametric or nonparametric form. Identifying the CIF and its associated parameters is essentially a neural encoding problem (Section 4.2). A convenient modeling framework is the generalized linear model (GLM) [14, 15], which can model the binary (0/1) or spike count measurements. Within the exponential family, one can use the `logit` link function to model the binomial distribution, which has a generic form of $\log(p_t/(1-p_t)) = \theta^{\top} X$; one can also use the `log` link function to model the Poisson distribution, which has a generic form of $\log(\lambda_t) = \theta^{\top} X$.

In addition, researchers have used the negative binomial distribution to model spike count observations to capture the overdispersion phenomenon (where the variance is greater than the mean statistic). In many cases, for the purpose of computational tractability, researchers often use a Gaussian approximation for Poisson spike counts through a variance stabilization transformation. Table 1 lists a few population probability distributions for modeling spike count observations.

Another popular statistical model for characterizing population spike trains is the maximum entropy (MaxEnt) model with a log-linear form [16, 17]. Given an ensemble of C neurons, the ensemble spike activity can be characterized by the following form:

$$\begin{aligned} p(X) &= \frac{1}{\mathcal{Z}(X)} \exp\left(\sum_{i=1}^{C} \theta_c \langle x_c \rangle + \sum_{i,j}^{C} \theta_{ij} \langle x_i x_j \rangle \right) \\ &\equiv \frac{1}{\mathcal{Z}(X)} \exp\left(\sum_{i=1}^{C+C^2} \theta_i f_i(X) \right), \end{aligned} \tag{9}$$

where $x_i \in \{-1, +1\}$, $\langle \cdot \rangle$ denotes the sample average, $\langle x_c \rangle$ denotes the mean firing rate of the cth neuron, $f_i(X)$ denotes a generic function of X (where the couplings θ_i have to match the measured expectation values $\langle f_i(X) \rangle$), and $\mathcal{Z}(X)$ denotes the partition function. The basic MaxEnt model (9) assumes the stationarity of the data and includes the first- and second-order moment statistics but no stimulus component, but these assumptions can be relaxed to further derive an extended model.

An important issue for characterizing neural spike trains is model selection and the associated goodness-of-fit assessment. For goodness-of-fit assessment of spike train models, the reader is referred to [11, 18]. In addition, standard statistical techniques such as cross-validation, leave-one-out, and the receiver-operating-characteristic (ROC) curve may be considered. The model selection issue can be resolved by the likelihood principle based on well-established criteria (such as the *Bayesian information criterion* or *Akaike's information criterion*) [9, 11] or resolved by the Bayesian principle. Bayesian model selection and variable selection will be reviewed in Section 3.4.

3. Bayesian Modeling and Inference Methods

The common strategy of Bayesian modeling is to start with specific prior distributions for the unknowns. The prior distributions are characterized by some hyperparameters, which can be directly optimized or modeled by the second-level hyperpriors. If the prior is conjugate to the likelihood, then the posterior has the same form as the prior [8]. Hierarchical Bayesian modeling characterizes the uncertainties of all unknowns at different levels.

In this section, we will review some, either exact or approximate, Bayesian inference methods. The approximate Bayesian inference methods aim to compute or evaluate the integration by approximation. There are two types of approaches to accomplish this goal: deterministic approximation and stochastic approximation. The deterministic approximation can rely on Gaussian approximation, deterministic sampling (e.g., sigma-point approximation [19, 20]) or variational approximation [21–23]. The stochastic approximation uses Monte Carlo sampling to achieve a point mass representation of the probability distribution. These two approaches have been employed to approximate the likelihood or posterior function in many inference problems, such as model selection, filtering and smoothing, and state and parameter joint estimation. Detailed coverage of these topics can be found in many excellent books (e.g., [24–28]).

3.1. Variational Bayes (VB). VB is based on the idea of variational approximation [21–23] and is also referred to as *ensemble learning* [24]. To avoid overfitting in maximum likelihood estimation, VB aims to maximize the marginal log-likelihood or its lower bound as follows:

$$\begin{aligned}\log p(Y) &= \log \int d\theta \int dX p(\theta)\, p(X, Y \mid \theta)\\ &= \log \int d\theta \int dX q(X,\theta) \frac{p(\theta)\, p(X, Y \mid \theta)}{q(X,\theta)}\\ &\geq \int d\theta \int dX q(X,\theta) \log \frac{p(\theta)\, p(X, Y \mid \theta)}{q(X,\theta)}\\ &= \left\langle \log p(X, Y, \theta)\right\rangle_q + \mathscr{H}_q(X,\theta) \equiv \mathscr{F}(q(X,\theta)),\end{aligned} \tag{10}$$

where $p(\theta)$ denotes the parameter prior distribution, $p(X, Y \mid \theta)$ defines the complete data likelihood, and $q(X,\theta)$ is called the variational posterior distribution which approximates the joint posterior of the unknown state and parameter $p(X, \theta \mid Y)$. The term $\mathscr{H}_q$ represents the entropy of the variational posterior distribution q, and $\mathscr{F}(q(X,\theta))$ is referred to as the free energy. The lower bound is derived based on the *Jensen's inequality* [29]. Maximizing the free energy $\mathscr{F}(q(X,\theta))$ is equivalent to minimizing the Kullback-Leibler (KL) divergence [29] between the variational posterior and true posterior (denoted by $\mathrm{KL}(q \parallel p)$); since the KL divergence is nonnegative, we have $\mathscr{F}(q) = \log p(Y) - \mathrm{KL}(q \parallel p) \leq \log p(Y)$. The optimization problem in (10) can be resorted to the VB-EM algorithm [23] in a similar fashion as the standard EM algorithm [30].

A common (but not necessary) VB assumption is a factorial form of the posterior $q(X,\theta) = q(X)q(\theta)$, although one can further impose certain structure within the parameter space. In the case of mean-field approximation, we have $q(X,\theta) = q(X)\prod_i q(\theta_i)$. With selected priors $p(X)$ and $p(\theta)$, one can maximize the free energy by alternatively solving two equations: $\partial\mathscr{F}/\partial X = 0$ and $\partial\mathscr{F}/\partial\theta = 0$. Specifically, VB-EM inference can be viewed as a natural extension of the EM algorithm, which consists of the following two steps.

(i) VB-E step: given the available information of $q(\theta)$, maximize the free energy $\mathscr{F}$ with respect to the function $q(X)$ and update the posterior $q(X)$.

(ii) VB-M step: given the available information of $q(X)$, maximize the free energy $\mathscr{F}$ with respect to the function $q(\theta)$ and update the posterior $q(\theta)$. The posterior update will have an analytic form provided that the prior $p(\theta)$ is conjugate to the complete-data likelihood function (the conjugate-exponential family).

These two steps are alternated repeatedly until the VB algorithm reaches the convergence (say, the incremental change of $\mathscr{F}$ value is below a small threshold). Similar to the iterative EM algorithm, the VB-EM inference has local maxima in optimization. To resolve this issue, one may use multiple random initializations or employ a deterministic annealing procedure [31]. The EM algorithm can be viewed as a variant of the VB algorithm in that the VB-M step replaces the point estimate (i.e., $q(\theta) = \delta(\theta - \theta_{\mathrm{MAP}})$) from the traditional M-step with a full posterior estimate. Another counterpart of the VB-EM is the maximization-expectation (ME) algorithm [32], in which the VB-E step uses the MAP point estimate $q(X) = \delta(X - X_{\mathrm{MAP}})$, while the VB-M step updates the full posterior.

It is noted that when the latent variables and parameters are intrinsically coupled or statistically correlated, the mean-field approximation will not be accurate, and consequently the VB estimate will be strongly biased. To alleviate this problem, the latent-space VB (LSVB) method [33, 34] aims to maximize a tighter lower bound of the marginal log-likelihood from (10) as follows:

$$\begin{aligned}\log p(Y) &\geq \int dX q(X) \log \frac{p(X, Y)}{q(X)}\\ &= \int dX q(X) \log \frac{\int d\theta p(X, Y, \theta)\, p(\theta)}{q(X)}\\ &\equiv \mathscr{F}(q(X)) \geq \max_{q(\theta)} \mathscr{F}(q(X)\, q(\theta)).\end{aligned} \tag{11}$$

The reader is referred to [33, 34] for more details and algorithmic implementation.

Note. (i) Depending on specific problems, the optimization bound of VB methods may not be tight, which may cause a large estimate bias or underestimated variance [35]. Desirably, a *data-dependent* lower bound is often tighter (such as the one used in Bayesian logistic regression [25]). (ii) It was shown in [36] that the VB method for statistical models with latent variables can be viewed as a special case of local variational approximation, where the log-sum-exp function is used to form the lower bound of the log-likelihood. (iii) The VB-EM inference was originally developed for the probabilistic models in the conjugate-exponential family, but it can be extended to more general models based on approximation [37].

3.2. *Expectation Propagation (EP).*

EP is a message-passing algorithm that allows approximate Bayesian inference for factor graphs (one type of probabilistic graphical model that shows how a function of several variables can be factored into a product of simple functions and can be used to represent a posterior distribution) [38]. For a specific r.v. X (either continuous or discrete), the probability distribution $p(X)$ is represented as a product of factors as follows:

$$p(X) = \prod_a f_a(X). \tag{12}$$

The basic idea of EP is to "divide-and-conquer" by approximating the factors one by one as follows:

$$f_a(X) \longrightarrow \tilde{f}_a(X) \tag{13}$$

and then use the product of approximated term as the final approximation as follows:

$$q(X) = \prod_a \tilde{f}_a(X). \tag{14}$$

As a result, EP replaces the global divergence $\mathrm{KL}(p(X) \parallel q(X))$ by the local divergence between two product chains as follows:

$$\begin{aligned} &\mathrm{KL}(p(X) \parallel q(X)) \\ &\quad = \mathrm{KL}\left(f_a(X)\prod_{b\neq a} f_b(X) \parallel \tilde{f}_a(X)\prod_{b\neq a}\tilde{f}_b(X)\right) \\ &\quad \approx \mathrm{KL}\left(f_a(X)\prod_{b\neq a}\tilde{f}_b(X) \parallel \tilde{f}_a(X)\prod_{b\neq a}\tilde{f}_b(X)\right). \end{aligned} \tag{15}$$

To minimize (15), the EP inference procedure is planned as follows.

Step 1. Use message-passing algorithms to pass messages $\tilde{f}_a(X)$ between factors.

Step 2. Given the received message $\tilde{f}_b(X)$ for factor a (for all $b \neq a$), minimize the local divergence to obtain $\tilde{f}_a(X)$, and send it to other factors.

Step 3. Repeat the procedure until convergence.

Note. (i) EP aims to find the closest approximation q such that $\mathrm{KL}(p \parallel q)$ is minimized, whereas VB aims to find the variational distribution to minimize $\mathrm{KL}(q \parallel p)$ (note that the KL divergence is asymmetric, and $\mathrm{KL}(p \parallel q)$ and $\mathrm{KL}(q \parallel p)$ have different geometric interpretations [39]). (ii) Unlike the global approximation technique (e.g., moment matching), EP uses a local approximation strategy to minimize a series of local divergence.

3.3. *Markov Chain Monte Carlo (MCMC).*

MCMC methods are referred to as a class of algorithms for drawing random samples from probability distributions by constructing a Markov chain that has the equilibrium distribution as the desired distribution [40]. The designed Markov chain is reversible and has detailed balance. For example, given a transition probability P, the detailed balance holds between each pair of state i and j in the state space if and only if $\pi_i P_{ij} = \pi_j P_{ji}$ (where $\pi_i = \Pr(X_{t-1} = i)$ and $P_{ij} = \Pr(X_{t-1} = i, X_t = j)$). The appealing use of MCMC methods for Bayesian inference is to numerically calculate high-dimensional integrals based on the samples drawn from the equilibrium distribution [41].

The most common MCMC methods are the random walk algorithms, such as the Metropolis-Hastings (MH) algorithm [42, 43] and Gibbs sampling [44]. The MH algorithm is the simplest yet the most generic MCMC method to generate samples using a random walk and then to accept them with a certain acceptance probability. For example, given a random-walk proposal distribution $g(Z \rightarrow Z')$ (which defines a conditional probability of moving state Z to Z'), the MH acceptance probability $\mathscr{A}(Z \rightarrow Z')$ is

$$\mathscr{A}(Z \longrightarrow Z') = \min\left(1, \frac{p(Z')\, g(Z' \longrightarrow Z)}{p(Z)\, g(Z \longrightarrow Z')}\right), \tag{16}$$

which gives a simple MCMC implementation. Gibbs sampling is another popular MCMC method that requires no parameter tuning. Given a high-dimensional joint distribution $p(Z) = p(z_1, \ldots, z_n)$, Gibbs sampler draws samples from the individual conditional distribution $p(z_i \mid Z_{-i})$ in turn while holding others fixed (where Z_{-i} denote the $n-1$ variables in Z except for z_i).

For high-dimensional sampling problems, the random-walk behavior of the proposal distribution may not be efficient. Imagine that there are two directions (increase or decrease in the likelihood space) for a one-dimensional search; there will be 2^n search directions in an n-dimensional space. On average, it will take about $2^n/n$ steps to hit the exact search direction. Notably, some sophisticated MCMC algorithms employ side information to improve the efficiency of the sampler (i.e., the "mixing" of the Markov chain). Examples of non-random-walk methods include successive over-relaxation, hybrid Monte Carlo, gradient-based Langevin MCMC, and Hessian-based MCMC [24, 45–47].

Many statistical estimation problems (e.g., change point detection, clustering, and segmentation) consist in identifying the unknown number of statistical objects (e.g., change points, clusters, and boundaries), which are categorized as the variable-dimensional statistical inference problem. For this kind of inference problem, the so-called reversible jump MCMC (RJ-MCMC) method has been developed [48], which can be viewed as a variant of MH algorithm that allows proposals to change the dimensionality of the space while satisfying the detailed balance of the Markov chain.

Note. As discussed in Section 2.2, since the fundamental operations of Bayesian statistics involve integration, the MCMC methods appear naturally as the most generic techniques for Bayesian inference. On the one hand, the recent decades have witnessed an exponential growth in the MCMC literature for its own theoretic and algorithmic developments. On the other hand, there has been also an increasing trend in applying MCMC methods to neural data analysis, ranging from spike sorting, tuning curve estimation, and neural decoding to functional connectivity analysis, some of which will be briefly reviewed in Section 4.

3.4. Bayesian Model Selection and Variable Selection. Statistical model comparison can be carried on by Bayesian inference. From Bayes' rule, the model posterior probability is expressed by

$$p\left(\mathcal{M}_i \mid D\right) \propto p\left(D \mid \mathcal{M}_i\right) p\left(\mathcal{M}_i\right). \tag{17}$$

Under the assumption of equal model priors, maximizing the model posterior is equivalent to maximizing the model evidence (or marginal likelihood) as follows:

$$\begin{aligned} p\left(D \mid \mathcal{M}_i\right) &= \int_\theta p\left(D, \theta \mid \mathcal{M}_i\right) d\theta \\ &= \int_\theta p\left(D \mid \theta, \mathcal{M}_i\right) p\left(\theta \mid \mathcal{M}_i\right) d\theta. \end{aligned} \tag{18}$$

The Bayes factor (BF), defined as the ratio of evidence between two models, can be computed as [49]

$$\begin{aligned} \mathrm{BF} &= \frac{p\left(D \mid \mathcal{M}_1\right)}{p\left(D \mid \mathcal{M}_2\right)} = \frac{\int p\left(D, \theta_1 \mid \mathcal{M}_1\right) d\theta_1}{\int p\left(D, \theta_2 \mid \mathcal{M}_2\right) d\theta_2} \\ &= \frac{\int p\left(\theta_1 \mid \mathcal{M}_1\right) p\left(D \mid \theta_1, \mathcal{M}_1\right) d\theta_1}{\int p\left(\theta_2 \mid \mathcal{M}_2\right) p\left(D \mid \theta_2, \mathcal{M}_2\right) d\theta_2}. \end{aligned} \tag{19}$$

Specifically, the BF is treated as the Bayesian alternative to P values for testing hypotheses (in model selection) and for quantifying the degree the observed data support or conflict with a hypothesis [50]. As discussed previously in Section 3.1, the marginal likelihood may be intractable for a large class of probabilistic models. In practice, the BF is often computed based on numerical approximation, such as the Laplace-Metropolis Estimator [51] or sequential Monte Carlo methods [52]. In addition, for a large sample size, the logarithm of the BF can be roughly approximated by the Bayesian information criterion (BIC) [9], whose computation is much simpler without involving numerical integration.

Bayesian model selection can also be directly implemented via the so-called MCMC model composition (MC^3). The basic idea of MC^3 is to simulate a Markov chain $\{\mathcal{M}(t)\}$ with an equilibrium distribution as $p(\mathcal{M}_i \mid D)$. For each model $\mathcal{M}$, define a neighborhood $\mathrm{nbd}(\mathcal{M})$ and a transition matrix q by setting $q(\mathcal{M} \rightarrow \mathcal{M}') = 0$ for all $\mathcal{M}' \notin \mathrm{nbd}(\mathcal{M})$. Draw a new sample $\mathcal{M}'$ from $q(\mathcal{M} \rightarrow \mathcal{M}')$ and accept the new sample with a probability

$$\min \left\{1, \frac{p\left(\mathcal{M}' \mid D\right)}{p\left(\mathcal{M} \mid D\right)}\right\}. \tag{20}$$

Otherwise the chain remains unchanged. Once the Markov chain converges to the equilibrium, one can construct the model posterior based on Monte Carlo samples.

Within a fixed model class, it is often desirable to have a compact or sparse representation of the model to alleviate overfitting. Namely, many coefficients of the model parameters are zeros. A very useful approach for variable selection is the so-called automatic relevance determination (ARD) that encourages sparse Bayesian learning [24, 26, 53]. More specifically, ARD provides a way to infer hyperparameters in hierarchical Bayesian modeling. Given the likelihood $p(Y \mid \theta)$ and the parameter prior $p(\theta \mid \omega)$ (where ω denotes the hyperparameters), one can assign a hyperprior $p(\omega \mid \eta)$ for ω such that the marginal distribution $p(\theta) = \int p(\theta \mid \omega) p(\omega) d\omega$ is peaked and long-tailed (thereby favoring a sparse solution). The hyperprior $p(\omega)$ can be either identical or different for each element in θ. In the most general form, we can write

$$p(\theta) = \prod_i p\left(\theta_i\right) = \prod_i \int p\left(\theta_i \mid \omega_i\right) p\left(\omega_i \mid \eta_i\right) d\omega_i. \tag{21}$$

The hyperprior parameters $\eta = \{\eta_i\}$ can be fixed or optimized from data. Upon completing Bayesian inference, the estimated mean and variance statistics of some coefficients θ_i will be close to zero (i.e., with the least relevance) and therefore can be truncated. The ARD principle has been widely used in various statistical models, such as linear regression, GLM, and the relevance vector machine (RVM) [26].

3.5. Bayesian Model Averaging (BMA). BMA is a statistical technique aiming to account for the uncertainty in the model selection process [54]. By averaging many different competing statistical models (e.g., linear or Cox regression and GLM), BMA incorporates model uncertainties into parameter inference and data prediction.

Consider an example of GLM involving choosing independent variables and the link function. Every possible combination of choices defines a different model, say $\{\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_K\}$ (where $\mathcal{M}_0$ denotes the null model). Upon computing K Bayes factors $\mathrm{BF}_{10} = p(D \mid \mathcal{M}_1)/p(D \mid \mathcal{M}_0)$, $\mathrm{BF}_{20} = p(D \mid \mathcal{M}_2)/p(D \mid \mathcal{M}_0), \ldots$, and $\mathrm{BF}_{K0} = p(D \mid \mathcal{M}_K)/p(D \mid \mathcal{M}_0)$, the posterior probability $p(\mathcal{M}_k \mid D)$ is computed as [54]

$$p\left(\mathcal{M}_k \mid D\right) = \frac{\pi_k \mathrm{BF}_{k0}}{\sum_{i=0}^{K} \pi_i \mathrm{BF}_{i0}}, \tag{22}$$

where $\pi_k = p(\mathcal{M}_k)/p(\mathcal{M}_0)$ denotes the prior odds for model $\mathcal{M}_k$ against $\mathcal{M}_0$. In the case of GLM, the marginal likelihood can be approximated by the Laplace method [55].

3.6. Bayesian Filtering: Kalman Filter, Point Process Filter, and Particle Filter. Bayesian filtering aims to infer a filtered or predictive posterior distribution of temporal data in a sequential fashion, which is often cast within the framework of state space model (SSM) [13, 56, 57]. Without loss of generality, let $\mathbf{x}_t$ denote the state at discrete time t and let $\mathbf{y}_{0:t}$ denote the cumulative observations up to time t. The filtered posterior distribution of the state, conditional on the observations $\mathbf{y}_{0:t}$, bears a form of *recursive* Bayesian estimation as follows:

$$\begin{aligned} p(\mathbf{x}_t \mid \mathbf{y}_{0:t}) &= \frac{p(\mathbf{x}_t)\,p(\mathbf{y}_{0:t} \mid \mathbf{x}_t)}{p(\mathbf{y}_{0:t})} \\ &= \frac{p(\mathbf{x}_t)\,p(\mathbf{y}_t, \mathbf{y}_{0:t-1} \mid \mathbf{x}_t)}{p(\mathbf{y}_t, \mathbf{y}_{0:t-1})} \\ &= \frac{p(\mathbf{x}_t)\,p(\mathbf{y}_t \mid \mathbf{x}_t, \mathbf{y}_{0:t-1})\,p(\mathbf{y}_{0:t-1} \mid \mathbf{x}_t)}{p(\mathbf{y}_t \mid \mathbf{y}_{0:t-1})\,p(\mathbf{y}_{0:t-1})} \\ &= \frac{p(\mathbf{x}_t)\,p(\mathbf{y}_t \mid \mathbf{x}_t, \mathbf{y}_{0:t-1})\,p(\mathbf{x}_t \mid \mathbf{y}_{0:t-1})\,p(\mathbf{y}_{0:t-1})}{p(\mathbf{y}_t \mid \mathbf{y}_{0:t-1})\,p(\mathbf{y}_{0:t-1})\,p(\mathbf{x}_t)} \\ &= \frac{p(\mathbf{y}_t \mid \mathbf{x}_t, \mathbf{y}_{0:t-1})\,p(\mathbf{x}_t \mid \mathbf{y}_{0:t-1})}{p(\mathbf{y}_t \mid \mathbf{y}_{0:t-1})} \\ &= \frac{p(\mathbf{y}_t \mid \mathbf{x}_t)\,p(\mathbf{x}_t \mid \mathbf{y}_{0:t-1})}{p(\mathbf{y}_t \mid \mathbf{y}_{0:t-1})}, \end{aligned} \tag{23}$$

where the first four steps are derived from Bayes' rule and the last equality of (23) assumes the conditional independence between the observations. The one-step state prediction, also known as the *Chapman-Kolmogorov equation* [58], is given by

$$p(\mathbf{x}_t \mid \mathbf{y}_{0:t-1}) = \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1})\,p(\mathbf{x}_{t-1} \mid \mathbf{y}_{0:t-1})\,d\mathbf{x}_{t-1}, \tag{24}$$

where the probability distribution (or density) $p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ describes a state transition equation and the probability distribution (or density) $p(\mathbf{y}_t \mid \mathbf{x}_t)$ is the observation equation. Together (23) and (24) provide the fundamental relations to conduct state space analyses. The above formulation of recursive Bayesian estimation holds for both continuous and discrete variables, for either $\mathbf{x}$ or $\mathbf{y}$ or both. When the state variable is discrete and countable (in which we use S_t to replace $\mathbf{x}_t$), the SSM is also referred to as a hidden Markov model (HMM), with associated $p(S_t \mid S_{t-1})$ and $p(\mathbf{y}_t \mid S_t)$. Various approximate Bayesian methods for the HMM have been reported [23, 59, 60]. When the hidden state consists of both continuous and discrete variables, the SSM is referred to as a switching SSM, with associated $p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, S_t)$ and $p(\mathbf{y}_t \mid \mathbf{x}_t, S_t)$ [27, 61]. In this case, the inference and prediction involve multiple integrals or summations. For example, the prediction equation (24) will be rewritten as

$$\begin{aligned} p(\mathbf{x}_t \mid \mathbf{y}_{0:t-1}, S_{0:t-1}) = \int \sum_{S_{t-1}} & p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, S_t)\,p(S_t \mid S_{t-1}) \\ & \times p(\mathbf{x}_{t-1} \mid \mathbf{y}_{0:t-1}, S_{0:t-1})\,d\mathbf{x}_{t-1} \end{aligned} \tag{25}$$

whose exact or naive implementation can be computationally prohibitive given a large discrete state space.

When the state and observation equations are both continuous and Gaussian, the Bayesian filtering solution yields the celebrated Kalman filter [62], in which the posterior mean and posterior variance are updated recursively. In fact, based on a Gaussian approximation of nonnegative spike count observations, the Kalman filter has been long used in spike train analysis [63, 64]. However, such a naive Gaussian approximation does not consider the point process nature of neural spike trains. Brown and his colleagues [65–67] have proposed a point process filter to recursively estimate the state or parameter in a dynamic fashion. Without loss of generality, assume that the CIF (6) is characterized by a parameter θ via an exponential form, namely, $\lambda_t \equiv \lambda(t \mid \theta_t) = \exp(\theta_t^\top X_t)$, and assume that the parameter follows a random-walk equation $\theta_t = \theta_{t-1} + w_t$ (where w_t denotes random Gaussian noise with zero mean and variance σ^2); then one can use a point process filter to estimate the time-varying parameter θ at arbitrarily fine temporal resolution (i.e., the bin size can be as small as possible for the discrete-time index t) as follows:

$$\theta_{t+1|t} = \theta_{t|t} \quad (\text{one-step mean prediction}), \tag{26}$$

$$V_{t+1|t}(\theta) = V_{t+1|t}(\theta) + \sigma^2 \quad (\text{one-step variance prediction}), \tag{27}$$

$$\begin{aligned} \theta_{t+1|t+1} &= \theta_{t+1|t} + V_{t+1|t}(\theta)\,\frac{\nabla_\theta \lambda(\theta_{t+1|t})}{\lambda(\theta_{t+1|t})} \\ &\quad \times [dy_{t+1} - \lambda(\theta_{t+1|t+1})\,\Delta] \\ &= \theta_{t+1|t} + V_{t+1|t}(\theta)\,X_{t+1} \\ &\quad \times [dy_{t+1} - \lambda(\theta_{t+1|t+1})\,\Delta] \quad (\text{posterior mode}), \end{aligned} \tag{28}$$

$$\begin{aligned} V_{t+1|t+1}(\theta) = \left[(V_{t+1|t}(\theta))^{-1} + X_{t+1}X_{t+1}^\top \lambda(\theta_{t+1|t})\,\Delta\right]^{-1} \\ (\text{posterior variance}), \end{aligned} \tag{29}$$

where $\theta_{t+1|t+1}$ and $V_{t+1|t+1}(\theta)$ denote the posterior mode and posterior variance for the parameter θ, respectively. Equations (26)–(29) are reminiscent of Kalman filtering. Equations (26) and (27) for one-step mean and variance predictions are the same as Kalman filtering, but (28) and (29) are different from Kalman filtering due to the presence of non-Gaussian observations and nonlinear operation in (28). In (28), $[dy_{t+1} - \lambda(\theta_{t+1|t+1})\Delta]$ is viewed as the innovations term, and $V_{t+1|t}X_{t+1}$ may be interpreted as a "Kalman gain." The quantity of the Kalman gain determines the "step size" in error correction. In (29), the posterior state variance is derived by inverting the second derivative of the log-posterior probability density $\log p(\theta_t \mid Y)$ based on a Gaussian approximation of the posterior distribution around the posterior mode [65–67]. For this simple example, we have

$$\begin{aligned} &\log p(\theta_t \mid Y_{0:t}, H_t) \\ &\quad \propto -\frac{1}{2}(\theta_t - \theta_{t-1|t-1})^\top V_{t+1|t}^{-1}(\theta_t - \theta_{t-1|t-1}) \\ &\qquad + [\log \lambda_t dy_t - \lambda_t \Delta], \end{aligned}$$

$$\begin{aligned}
\frac{\partial \log p\left(\theta_t \mid Y_{0:t}, H_t\right)}{\partial \theta_t}
&= -\left(\theta_t - \theta_{t-1|t-1}\right)^{\top} V_{t+1|t}^{-1} \\
&\quad + \frac{1}{\lambda_t} \nabla_{\theta} \lambda_t \left[dy_t - \lambda_t \Delta\right], \\
\frac{\partial^2 \log p\left(\theta_t \mid Y_{0:t}, H_t\right)}{\partial \theta_t \partial \theta_t^{\top}}
&= -V_{t+1|t}^{-1} + \left[\left(\frac{\partial^2 \lambda_t}{\partial \theta_t \partial \theta_t^{\top}} \frac{1}{\lambda_t} - \left(\frac{\partial \lambda_t}{\partial \theta_t} \right)^2 \frac{1}{\lambda_t^2} \right) \right. \\
&\qquad \left. \times \left[dy_t - \lambda_t \Delta\right] - \left(\frac{\partial \lambda_t}{\partial \theta_t} \right)^2 \frac{1}{\lambda_t} \Delta \right].
\end{aligned} \tag{30}$$

Setting the first-order derivative $\partial \log p(\theta_t \mid Y_{0:t}, H_t)/\partial \theta_t$ to zero and rearranging terms yield (28), and setting $V_{t+1|t+1}(\theta) = -[\partial^2 \log p(\theta_t \mid Y_{0:t}, H_t)/(\partial \theta_t \partial \theta_t^{\top})]^{-1}$ yields (29).

The Gaussian approximation is based on the first-order Laplace method. In theory one can also use a second-order method to further improve the approximation accuracy [68]. However, in practice the performance gain is relatively small in the presence of noise and model uncertainty while analyzing real experimental data sets. Although the above example only considers a univariate point process (i.e., a single neuronal spike train), it is straightforward to extend the analysis to multivariate point processes (multiple neuronal spike trains). When the number of the neurons increases, the accuracy of Gaussian approximation of log-posterior also improves due to the *Law of large numbers*.

An alternative way for estimating a non-Gaussian posterior is to use a particle filter [69]. Several reports have been published in the context of neural spike train analysis [70, 71]. The basic idea of particle filtering is to employ sequential Monte Carlo (importance sampling and resampling) methods and draw a set of independent and identically distributed (i.i.d.) samples (i.e., "particles") from a *proposal distribution*; the samples are propagated through the likelihood function, weighted, and reweighted after each iteration update. In the end, one can use Monte Carlo samples (or their importance weights) to represent the posterior. For example, to evaluate the expectation of a function $f(\mathbf{x}_t)$ with respect to the posterior $p(\mathbf{x}_t \mid \mathbf{y}_{0:t})$, we have

$$\begin{aligned}
\mathbb{E}\left[f\left(\mathbf{x}_t\right)\right] &= \int f\left(\mathbf{x}_t\right) \frac{p\left(\mathbf{x}_t \mid \mathbf{y}_{0:t}\right)}{q\left(\mathbf{x}_t \mid \mathbf{y}_{0:t}\right)} q\left(\mathbf{x}_t \mid \mathbf{y}_{0:t}\right) d\mathbf{x}_t \\
&= \int f\left(\mathbf{x}_t\right) W\left(\mathbf{x}_t\right) q\left(\mathbf{x}_t \mid \mathbf{y}_{0:t}\right) d\mathbf{x}_t \\
&\approx \frac{\sum_{i=1}^{N_c} f\left(\mathbf{x}_t^{(i)}\right) W\left(\mathbf{x}_t^{(i)}\right)}{\sum_{i=1}^{N_c} W\left(\mathbf{x}_t^{(i)}\right)} = \widehat{f}\left(\mathbf{x}_t\right),
\end{aligned} \tag{31}$$

where $W(\mathbf{x}_t) = p(\mathbf{x}_t \mid \mathbf{y}_{0:t})/q(\mathbf{x}_t \mid \mathbf{y}_{0:t})$ denotes the importance weight function and $\{\mathbf{x}_t^{(i)}\}_{i=1}^{N_c}$ denotes the N_c particles drawn from the proposal distribution $q(\mathbf{x}_t \mid \mathbf{y}_{0:t})$. When the sample size N_c is sufficiently large (depending on the dimensionality of $\mathbf{x}$), the estimate $\widehat{f}(\mathbf{x}_t)$ will be an unbiased estimate of $\mathbb{E}[f(\mathbf{x}_t)]$. Based on sequential important sampling (SIS), the importance weights of each sample can be recursively updated as follows [69]:

$$W\left(\mathbf{x}_t^{(i)}\right) = W\left(\mathbf{x}_{t-1}^{(i)}\right) \frac{p\left(\mathbf{y}_t \mid \mathbf{x}_t^{(i)}\right) p\left(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)}\right)}{q\left(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{0:t-1}^{(i)}, \mathbf{y}_t\right)}. \tag{32}$$

In practice, choosing a proper proposal distribution $q(\mathbf{x}_t \mid \mathbf{x}_{0:t-1}, \mathbf{y}_t)$ is crucial (see [69] for detailed discussions). In the neuroscience literature, Brockwell et al. [70] used a transition prior $p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ as the proposal distribution, which yields a simple form of update from (32) as follows:

$$W\left(\mathbf{x}_t^{(i)}\right) = W\left(\mathbf{x}_{t-1}^{(i)}\right) p\left(\mathbf{y}_t \mid \mathbf{x}_t^{(i)}\right). \tag{33}$$

That is, the importance weights $W(\mathbf{x}_t^{(i)})$ are only scaled by the instantaneous likelihood value. Despite its simplicity, the transition prior proposal distribution completely ignores the information of current observation $\mathbf{y}_t$. To overcome this limitation, Ergun et al. [71] used a filtered (Gaussian) posterior density derived from the point process filter as the proposal distribution, and they reported a significant performance gain in estimation while maintaining the algorithmic simplicity (i.e., sampling from a Gaussian distribution). In addition, the VB approach can be integrated with particle filtering to obtain a variational Bayesian filtering algorithm [72].

Note. (i) If the online operation is not required, we can estimate a smoothed posterior distribution $p(\mathbf{x}_t \mid \mathbf{y}_{0:T})$ to obtain a more accurate estimate. The above Bayesian filters can be extended to the fixed-lag Kalman smoother, point process smoother, and particle smoother [63, 66, 69]. (ii) For neural spike train analysis, the formulation of Bayesian filtering is applicable not only to simple point processes but also to marked point processes [73] or even spatiotemporal point processes.

3.7. Bayesian Nonparametrics.

The contrasting methodological pairs "*frequentist* versus *Bayes*" and "*parametric* versus *nonparametric*" are two examples of dichotomy in modern statistics [74]. The historical roots of Bayesian nonparametrics are dated back to the late 1960s and 1970s. Despite its theoretic development over the past few decades, successful applications of nonparametric Bayesian inference have not been widespread until recently, especially in the field of machine learning [75]. Since nonparametric Bayesian models accommodate a large number of degrees of freedom (infinite-dimensional parameter space) to exhibit a rich class of probabilistic structure, such approaches are very powerful in terms of data representation. The fundamental building blocks are two stochastic processes: Dirichlet process (DP) and Gaussian process (GP). Although detailed technical reviews of these topics are far beyond the scope of this paper, we would like to point out the strengths of these methods in two aspects of statistical data analysis.

(i) Data clustering, partitioning, and segmentation: unlike the finite mixture models, nonparametric

Bayesian models define a prior distribution over the set of all possible partitions, in which the number of clusters or partitions may grow as the increase of the data samples in both static and dynamic settings, including the infinite Gaussian mixture model, Dirichlet process mixtures, Chinese restaurant process, and infinite HMM [74–76]. The model selection issue is resolved implicitly in the process of infinite mixture modeling.

(ii) Prediction and smoothing: unlike the fixed finite-dimensional parametric models, the GP defines priors for the mean function and covariance function, where the covariance kernel function determines the smoothness and stationarity between the data points. Since the predictive posterior is Gaussian, the prediction uncertainty can be computed analytically [28, 77].

Therefore, Bayesian nonparametrics offer greater flexibility for modeling complex data structures. Unfortunately, most inference algorithms for Bayesian nonparametric models involve MCMC methods, which can be computationally prohibitive for large-scale neural data analysis. Therefore, exploiting the sparsity structure of specific neural data and designing efficient inference algorithms are two important directions in practical applications [78].

4. Bayesian Methods for Neural Spike Train Analysis

In this section, we review some representative applications of Bayesian methods for neural spike train analysis, with specific emphases on the real experimental data. Nevertheless, the list of the references is by no means complete, and some other complementary references can be found in [79, 80]. Specifically, the strengths of the Bayesian methods are highlighted in comparison with other standard methods; the potentially issues arising from these methods are also discussed.

4.1. Spike Sorting and Tuning Curve Estimation. To characterize the firing property of single neurons, it is necessary to first *identify* and *sort* the spikes from the recorded multiunit activity (MUA) (which is referred to as the discrete ensemble spikes passing the threshold criterion) [81–83]. However, spike sorting is often a difficult and error-prone process. Traditionally, spike sorting is formulated as a clustering problem based on spike waveform features [84]. Parametric and nonparametric Bayesian inference methods have been developed for mixture modeling and inference (e.g., [25, 26]), especially for determining the model size [85, 86]. Unlike the maximum likelihood estimation (which produces a hard label for each identified spike), Bayesian approaches produce a soft label (posterior probability) for individual spike; such uncertainties may be considered in subsequent analyses (such as tuning curve estimation and decoding). Spike sorting can also be formulated as a dynamic model inference problem, in the context of state space analysis [87] or in the presence of nonstationarity [88]. Recent studies have suggested that spike sorting should take into account not only spike waveform features but also the neuronal tuning property [89, 90], suggesting that these two processes shall be integrated.

At the single neuron level, a Poisson neuronal firing response is completely characterized by its tuning curve or receptive field (RF). Naturally, estimating the neuronal tuning curve is the second step following spike sorting. Standard tuning curve or RF estimation methods include the spike-triggered average (STA) and spike-triggered covariance (STC). The Bayesian versions of the STA and STC have been proposed [91, 92]. Binning and smoothing are two important issues in firing rate estimation Bayesian methods provide a principled way to estimate the peristimulus time histogram (PSTH) [93]. For estimating a time-varying firing rate profile similar to PSTH, the Bayesian adaptive regression splines (BARS) method offers a principled solution for bin size selection and smoothing based on the RJ-MCMC method [94]. Notably, BARS is more computationally intensive. For similar estimation performance (validated on simulated data), a more computationally efficient approach has been developed using Bayesian filtering-based state space analysis [95]. In addition, Metropolis-type MCMC approaches have been proposed for high-dimensional tuning curve estimation [96, 97].

4.2. Neural Encoding and Decoding. The goal of neural encoding is to establish a statistical mapping (which can be either a biophysical or data-driven model) between the stimulus input and neuronal responses, and the goal of neural decoding is to extract or reconstruct information of the stimulus given the observed neural signals. For instance, the encoded and decoded variables of interest can be a rodent's position during spatial navigation, the monkey's movement kinematics in a reach-to-grasp task, or specific visual/auditory/olfactory stimuli during neuroscience experiments.

Without loss of generality, let $\{\widetilde{X}, \widetilde{Y}\}$ denote the *observed* stimuli and neuronal responses, respectively, at the encoding stage, and let θ denote the model parameter of a specific encoding model $\mathscr{M}$; then the posterior distribution of the model (and model parameters) is written as

$$p\left(\theta, \mathscr{M} \mid \widetilde{X}, \widetilde{Y}\right) \propto p\left(\widetilde{X}, \widetilde{Y} \mid \theta, \mathscr{M}\right) p(\theta \mid \mathscr{M}) p(\mathscr{M}). \quad (34)$$

Once the model $\mathscr{M}$ is determined, one can infer the posterior mean by $\widehat{\theta} = \int \theta p(\theta \mid \widetilde{X}, \widetilde{Y}, \mathscr{M}) d\theta$. Depending on the selected likelihood or prior, variations of Bayesian neural encoding methods have been developed [98–100].

Given the parameter posterior $p(\theta \mid \widetilde{X}, \widetilde{Y}, \mathscr{M})$ from the encoding analysis, decoding analysis aims to infer the latent variable X given new data Y at the decoding stage (with preselected $\mathscr{M}$). Within the Bayesian framework, it is equivalent to finding the X_{MAP} [101] as follows:

$$\begin{aligned} X_{\mathrm{MAP}} &= \arg\max_{X} p(X \mid \theta, Y, \mathscr{M}) \\ &= \arg\max_{X} \int p(Y \mid X, \theta, \mathscr{M})\, p\left(\theta \mid \widetilde{X}, \widetilde{Y}, \mathscr{M}\right) p(X)\, d\theta \\ &\approx \arg\max_{X} p\left(Y \mid X, \widehat{\theta}, \mathscr{M}\right) p(X), \end{aligned} \quad (35)$$

which consists of two numerical problems: *maximization* and *integration*. In the approximation in the last step of (35), we have used $p(\theta \mid \widetilde{X}, \widetilde{Y}, \mathscr{M}) \approx \delta(\theta - \widehat{\theta})$, where $\widehat{\theta}$ denotes the estimated mean or mode statistic from $p(\theta \mid \widetilde{X}, \widetilde{Y}, \mathscr{M})$. The optimization problem is more conveniently written in the log domain as follows:

$$\log p\left(X \mid Y, \widehat{\theta}\right) \propto \log p\left(Y \mid X, \widehat{\theta}\right) + \log p(X). \qquad (36)$$

If X follows a Markovian process, this can be solved by recursive Bayesian filtering [65, 67] (Section 3.6). When X is non-Markovian but $p(X)$ and the likelihood are both log-concave, this can be resorted to a global optimization problem [57, 102]. Imposing prior information and structure (e.g., sparsity, spatiotemporal correlation) onto $p(X)$ is important for obtaining either a meaningful solution or a significant optimization speedup [103, 104]. In contrast, when $p(X)$ is flat or noninformative, the MAP solution will be similar to the m.l.e.

In the literature, the majority of neural encoding or decoding models fall within two parametric families: linear model (e.g., [63, 105]) and GLM (e.g., [64, 106, 107]), although nonparametric encoding models have also been considered [108, 109]. Methods for Bayesian neural decoding include (i) Kalman filtering [63], (ii) point process filtering [65–67, 110, 111], (iii) particle filtering [70, 71], and (iv) MCMC methods [112]. The areas of experimental neuroscience data include the retina, primary visual cortex, primary somatosensory cortex, auditory periphery (auditory nerves and midbrain auditory neurons), primary auditory cortex, primary motor cortex, premotor cortex, hippocampus, and the olfactory bulb.

It is important to point out that most spike-count or point process based decoding algorithms rely on the assumptions that neural spikes have been properly sorted (some neural decoding algorithms (e.g., [113]) are based on detected MUA instead of sorted single unit activity). Recently, there have been a few efforts in developing spike-sorting-free decoding algorithms, by either estimating the cell identities as missing variables [114] or modeling the spike identities by their proxy based on a spatiotemporal point process [115, 116]. Although this work has been carried out using likelihood inference, it is straightforward to extend it to the Bayesian framework. In the example of decoding the rat's position from recorded ensemble hippocampal spike activity [115, 116], we used a model-free (without θ) and data-driven Bayes' rule as follows:

$$p\left(X \mid Y, \widetilde{X}, \widetilde{Y}\right) \propto p\left(Y \mid X, \widetilde{X}, \widetilde{Y}\right) p(X), \qquad (37)$$

in which $p(X)$ denotes the prior and the likelihood $p(Y \mid X, \widetilde{X}, \widetilde{Y})$ is evaluated nonparametrically (namely, nonparametric neural decoding). By assuming that the joint/marginal/conditional distributions ($p(X, Y)$ and $p(\widetilde{X}, \widetilde{Y})$, $p(X)$ and $p(\widetilde{X})$, and $p(Y \mid X)$ and $p(\widetilde{Y} \mid \widetilde{X})$) are stationary during both encoding and decoding phases, the MAP estimate of decoding analysis is obtained by

$$\begin{aligned} X_{\mathrm{MAP}} &= \arg\max_X p\left(Y \mid X, \widetilde{X}, \widetilde{Y}\right) p(X) \\ &\approx \arg\max_X f\left(Y \middle| p\left(X \mid \widetilde{X}\right), p\left(X, Y \mid \widetilde{X}, \widetilde{Y}\right)\right) p(X), \end{aligned} \qquad (38)$$

where f is a nonlinear function that involves the marginal and joint pdf's in the argument [115, 116], in which the pdf's are constructed based on a kernel density estimator (KDE). Alternatively, the nonparametric pdf in (38) can be replaced by a parametric form [115] as follows:

$$X_{\mathrm{MAP}} \approx \arg\max_X f\left(Y \middle| p(X \mid \theta), p(X, Y \mid \theta)\right) p(X), \qquad (39)$$

where $p(X \mid \theta) = \int p(X, Y \mid \theta) dY$ is the parametric marginal and θ is the point estimate obtained from the training samples $\{\widetilde{X}, \widetilde{Y}\}$.

Note. (i) Neural encoding and decoding analyses are established upon the assumption that the neural codes are well understood—namely, how neuronal spikes represent and transmit the information of the external world. Whether being a rate code, a timing code, a latency code, or an independent or correlated population code, Bayesian approach provides a universal strategy to test the coding hypothesis or extract the information [117]. (ii) The sensitivity of spike trains to noise may affect the effectiveness to the encoding-decoding process. From an information-theoretic perspective, various sources of spike noise, such as misclassified spikes (false positives) and misdetected, or misclassified spikes (false negatives), may affect differently the mutual information between the input (stimulus) and output (spikes) channel [118, 119]. In designing a Bayesian decoder, it is important to take into account the noise issue. A decoding strategy that is robust to the noise assumption will presumably yield the best performance [115, 116].

4.3. Deconvolution of Neural Spike Trains. Fluorescent calcium imaging tools have become increasingly popular for observing the spiking activity of large neuronal populations. However, extracting or deconvolving neural spike trains from the raw fluorescence movie or video sequences remains a challenging estimation problem. The standard dF/F or Wiener filtering approaches do not capture the true statistics of neural spike trains and are sensitive to the noise statistics [120].

A principled approach is to formulate the deconvolution problem of a filtered point process via state space analysis and Bayesian inference [121, 122] (see also [123] for another type of Bayesian deconvolution approach using MCMC). Let F_t denote the measured univariate fluorescence time series, which is modeled as a linear Gaussian function of the intracellular calcium concentration ($[\mathrm{Ca}^{2+}]$) as follows:

$$F_t = \alpha\left[\mathrm{Ca}^{2+}\right]_t + \beta + \epsilon_t, \qquad (40)$$

where β denotes a constant baseline and $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$ denotes the Gaussian noise with zero mean and variance σ^2. The calcium concentration can be modeled as a first-order autoregressive (AR) process corrupted by Poisson noise as follows:

$$\alpha\left[\mathrm{Ca}^{2+}\right]_t = \alpha\left[\mathrm{Ca}^{2+}\right]_{t-1} + n_t, \tag{41}$$

where $n_t \sim \texttt{Poisson}\,(\lambda\Delta)$ and the bin size Δ is chosen to assure that the mean firing rate is independent of the imaging frame rate.

Let $\theta = \{\alpha, \beta, \gamma, \sigma^2, \lambda\}$; given the above generative biophysical model, Bayesian deconvolution aims to seek the MAP estimate of spike train as follows:

$$\begin{aligned}
\hat{\mathbf{n}} &= \arg\max_{n_t \in \mathbb{N}_0} p\left(\mathbf{n} \mid \mathbf{F}, \theta\right) \\
&= \arg\max_{n_t \in \mathbb{N}_0} p\left(\mathbf{F} \mid \mathbf{n}, \theta\right) p\left(\mathbf{n} \mid \theta\right) \\
&= \arg\max_{n_t \in \mathbb{N}_0} \prod_{t=1}^{T} p\left(F_t \mid \mathrm{Ca}_t^{2+}, \theta\right) \prod_{t=1}^{T} p\left(n_t \mid \theta\right).
\end{aligned} \tag{42}$$

Within the state space framework, Vogelstein and colleagues [121] proposed a particle filtering method to infer the posterior probability of spikes at each imaging frame, given the entire fluorescence traces. However, the Monte Carlo approach is computationally expensive and may not be suitable for analyses of a large population of neurons. To meet the real-time processing requirement, they further proposed an approximate yet fast solution by replacing the Poisson distribution by an exponential distribution with the same mean (therefore relaxing the nonnegative integer constraint to the nonnegative real number) [122]. And the approximate solution is given by the following optimization problem:

$$\begin{aligned}
\hat{\mathbf{n}} &= \arg\max_{n_t \geq 0} \sum_{t=1}^{T} -\frac{1}{2\sigma^2}\left(F_t - \alpha \mathrm{Ca}_t^{2+} - \beta\right)^2 - n_t \lambda\Delta \\
&= \arg\max_{\mathrm{Ca}_t^{2+} - \gamma \mathrm{Ca}_{t-1}^{2+} \geq 0} \sum_{t=1}^{T} -\frac{1}{2\sigma^2}\left(F_t - \alpha \mathrm{Ca}_t^{2+} - \beta\right)^2 \\
&\qquad - \left(\mathrm{Ca}_t^{2+} - \gamma \mathrm{Ca}_{t-1}^{2+}\right) \lambda\Delta.
\end{aligned} \tag{43}$$

The approximation of exponential form makes the optimization problem concave with respect to Ca^{2+}, from which the global optimum can be obtained using constrained convex optimization [102]. Once the estimate of the calcium trace is obtained, the MAP spike train can be simply inferred by a linear transformation.

In a parallel fashion, the parameter θ can be similarly estimated by Bayesian inference as follows:

$$\begin{aligned}
\theta_{\mathrm{MAP}} &= \arg\max_{\theta} \int p\left(\mathbf{F} \mid \mathrm{Ca}^{2+}, \theta\right) p\left(\mathrm{Ca}^{2+} \mid \theta\right) d\mathrm{Ca}^{2+} \\
&\approx \arg\max_{\theta} p\left(\mathbf{F} \mid \hat{\mathbf{n}}, \theta\right) p\left(\hat{\mathbf{n}} \mid \theta\right),
\end{aligned} \tag{44}$$

where the approximation in the second step assumes that the major mass in the integral is around the MAP sequence $\hat{\mathbf{n}}$ (or equivalently the Ca^{2+} traces). Therefore, the joint estimate $(\hat{\mathbf{n}}, \theta_{\mathrm{MAP}})$ can be computed iteratively from (43) and (44) until convergence.

Note. The output of Bayesian deconvolution yields a probability vector between 0 and 1 of having a spike in a given time frame. Selection of different thresholds on the probability vector leads to different detection errors (a tradeoff between the false positives and false negatives). Nevertheless, the Bayesian solution is much more superior to the standard least-squares method. It is noteworthy that a new fast deconvolution method has recently been proposed based on finite rate of innovation (FRI) theory, with reported performance better than the approximate Bayesian solution [124].

4.4. Inference of Neuronal Functional Connectivity and Synchrony.

Identifying the functional connectivity of simultaneously recorded neuronal ensembles is an important research objective in computational neuroscience. This analysis has many functional applications such as in neural decoding [125] and in understanding the collective dynamics of coordinated spiking cortical networks [126]. Compared to the standard nonparametric approaches such as cross-correlogram and joint peristimulus time histogram (JPSTH), parametric model-based statistical approaches offer several advantages in neural data interpretation [127].

To model the spike train point process data, without loss of generality we use the following logistic regression model with a logit link function. Specifically, let c be the index of a target neuron, and let $i = 1, \ldots C$ be the indices of triggered neurons (whose spike activity is assumed to trigger the firing of the target neuron). The Bernoulli (binomial) logistic regression GLM is written as

$$\begin{aligned}
\operatorname{logit}\left(\pi_t\right) &= \theta_c^{\top} X_t = \theta_0^c + \sum_{j=1}^{J} \theta_j^c x_{j,t} \\
&= \theta_0^c + \sum_{i=1}^{C} \sum_{k=1}^{K} \theta_{i,k}^c x_{i,t-k},
\end{aligned} \tag{45}$$

where $\dim(\theta_c) = J + 1 = C \times K + 1$ for the augmented parameter vector $\theta_c = \{\theta_0^c, \theta_{i,k}^c\}$ and $X_t = \{x_0, x_{i,t-k}\}$. Here, $x_0 \equiv 1$, and $x_{i,t-k}$ denotes the raw spike count from neuron i at the kth time-lag history window (or a predefined smooth basis function such as in [125]). The spike count is nonnegative; therefore $x_{i,t-k} \geq 0$. Alternatively, (45) can be rewritten as

$$\pi_t = \frac{\exp\left(\theta_c^{\top} X_t\right)}{1 + \exp\left(\theta_c^{\top} X_t\right)} = \frac{\exp\left(\theta_0^c + \sum_{j=1}^{J} \theta_j^c x_{j,t}\right)}{1 + \exp\left(\theta_0^c + \sum_{j=1}^{J} \theta_j^c x_{j,t}\right)}, \tag{46}$$

which yields the probability of a spiking event at time t. Equation (46) defines a spiking probability model for neuron c based on its own spiking history and that of the other neurons in the ensemble. Here, $\exp(\theta_0^c)$ can be interpreted as the baseline firing probability of neuron c. Depending on the algebraic (positive or negative) sign of coefficient $\theta_{i,k}^c$, $\exp(\theta_{i,k}^c)$ can be viewed as a "gain" factor (dimensionless, >1 or <1) that influences the relative firing probability of neuron c

from another neuron i at the previous kth time lag. Therefore, a negative value of $\theta_{i,k}^{c}$ will strengthen the inhibitory effect; a positive value of $\theta_{i,k}^{c}$ will enhance the excitatory effect. Two neurons are said to be functionally connected if any of their pairwise connections is nonzero (or the statistical estimate is significantly nonzero).

For inferring the functional connectivity of neural ensembles, in addition to the standard likelihood approaches [127, 128], various forms of Bayesian inference have been developed for the MaxEnt model, GLM, and Bayesian network [129–132]. In a similar context, a Bayesian method has been developed based on the deconvolved neuronal spike trains from calcium imaging data [133].

Bayesian methods also proved useful in detecting higher-order correlations among neural assemblies [134, 135]. Higher-order correlations are often characterized by synchronous neuronal firing at a timescale of 5–10 ms. These findings have been reported in experimental data from prefrontal cortex, somatosensory cortex, and visual cortex across many species and animals. Consider a set of C neurons. Each neuron is represented by two states: 1 (firing) or 0 (silent). At any time instant, the state of the C neurons is represented by the vector $X = (x_1, x_2, \ldots, x_C)$ (the time index is omitted for simplicity), and in total there are 2^C neuronal states. For instance, a general joint distribution of three neurons can be expressed by a log-linear model [134]

$$\begin{aligned} p(x_1, x_2, x_3) = \exp\big(&\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 \\ &+ \theta_{12} x_1 x_2 + \theta_{13} x_1 x_3 \\ &+ \theta_{23} x_2 x_3 + \theta_{123} x_1 x_2 x_3\big), \end{aligned} \tag{47}$$

which is a natural extension of the MaxEnt model described in (9). A nonzero coefficient of θ_{123} would imply the presence of third-order correlation among the three neurons. In experimental data, the number of synchronous events may be scarce in single trials, and the interaction coefficients may also be time-varying. State space analysis and Bayesian filtering offer a principled framework to address these issues [135]. However, the computational bottleneck is the curse of dimensionality when the value of C is moderately large ($2^{20} \approx 10^6$). In the presence of finite data sample size, it is reasonable to impose certain structural priors onto the parameter space for the Bayesian solution.

5. Discussion

We have presented an overview of Bayesian inference methods and their applications to neural spike train analysis. Although the focus of current paper is on neural spike trains, the Bayesian principle is also applicable to other modalities of neural data (e.g., [136]). Due to space limitation, we only cover representative methods and applications in this paper, and the references are reflective of our personal choices from the humongous literature.

In comparison with the standard methods, Bayesian methods provide a flexible framework to address many fundamental estimation problems at different stages of neural data analysis. Regardless of the specific Bayesian approach to be employed, the common goal of Bayesian solutions consists in replacing a single point estimate (or hard decision label) with a full posterior distribution (or soft decision label). As a tradeoff, Bayesian practioners have to encounter the increasing cost of computational complexity (especially while using MCMC), which may be prohibitive for large-scale spike train data sets. Furthermore, special attention shall be paid to select the optimal technique among different Bayesian methods that ultimately lead to quantitatively different approximate Bayesian solutions.

Despite the significant progresses made to date, there remain many research challenges and opportunities for applying Bayesian machinery to neural spike trains, and we will mention a few of them below.

5.1. Nonstationarity. Neural spiking activity is highly nonstationary at various timescales. Sources that account for such nonstationarity may include the animal's behavioral variability across trials, top-down attention, learning, motivation, or emotional effects across time. These effects are time-varying across behaviors. In addition, individual neuronal firing may be affected by other unobserved neural activity, such as through modulatory or presynaptic inputs from other nonrecorded neurons. Therefore, it may be important to consider these latent variables while analyzing neural spike trains [137]. Bayesian methods are a natural solution to model and infer such latent variables. Traditional mixed-effects models can be adapted to a hierarchical Bayesian model to capture various sources of randomness.

5.2. Characterization of Neuronal Dependencies. Neural responses may appear correlated or synchronous at different timescales. It is important to characterize such neuronal dependencies in order to fully understand the nature of neural codes. It is also equally important to associate the neural responses to other measurements, such as behavioral responses, learning performance, or local field potentials. Commonly, correlation statistics or information-theoretic measures have been used (e.g., [138]). Other advanced statistical measures have also been proposed, such as the log-linear model [139], Granger causality [140], transfer entropy [141], or copula model [142]. Specifically, the copula offers a universal framework to model statistical dependencies among continuous, discrete, or mixed-valued r.v., and it has an intrinsic link to the mutual information; Bayesian methods may prove useful for selecting the copula class or the copula mixtures [143]. However, because of the nonstationary nature of neural codes (Section 5.1), it remains a challenge to identify the "true" dependencies among the observed neural spike trains, and it remains important to rule out and rule in neural codes under specific conditions.

5.3. Characterization and Abstraction of Neuronal Ensemble Representation. Since individual neuronal spike activity is known to be stochastic and noisy, in the single-trial analysis it is anticipated that the information extracted from neuronal populations is more robust than that from a single neuron. How to uncover the neural representation of population codes in a single-trial analysis has been an active research topic in neuroscience. This is important not only for abstraction, interpretation, and visualization of population codes but also for discovering invariant neural representations and their links to behavior. Standard dimensionality reduction techniques (e.g., principle component analysis, multidimensional scaling, or locally linear embedding) have been widely used for such analyses. However, these methods have ignored the temporal component of neural codes. In addition, no explicit behavioral correlate may become available in certain modeling tasks. Recently, Bayesian dynamic models, such as the Gaussian process factor analysis (GPFA) [144] and VB-HMM [145–147], have been proposed to visualize population codes recorded from large neural ensembles across different experimental conditions. To learn the highly complex structure of spatiotemporal neural population codes, it may be beneficial to borrow the ideas from the machine learning community and to integrate the state-of-the-art unsupervised and supervised deep Bayesian learning techniques.

5.4. Translational Neuroscience Applications. Finally in the long run, it is crucial to apply basic neuroscience knowledge derived from quantitative analyses of neural data to translational neuroscience research. Many clinical research areas may benefit from the statistical analyses reviewed here, such as design of neural prosthetics for patients with tetraplegia [107], detection and control of epileptic seizures, optical control of neuronal firing in behaving animals, or simulation of neural firing patterns to achieve optimal electrotherapeutic effect [148]. Bridging the gap between neural data analysis and their translational applications (such as treating neurological or neuropsychiatric disorders) would continue to be a prominent mission accompanying the journey of scientific discovery.

Acknowledgments

The author was supported by an Early Career Award from the Mathematical Biosciences Institute, Ohio State University. This work was also supported by the NSF-IIS CRCNS (Collaborative Research in Computational Neuroscience) Grant (no. 1307645) from the National Science Foundation.

References

[1] E. N. Brown, R. E. Kass, and P. P. Mitra, “Multiple neural spike train data analysis: state-of-the-art and future challenges,” *Nature Neuroscience*, vol. 7, no. 5, pp. 456–461, 2004.

[2] S. Grün and S. Rotter, *Analysis of Parallel Spike Trains*, Springer, New York, NY, USA, 2010.

[3] I. H. Stevenson and K. P. Kording, “How advances in neural recording affect data analysis,” *Nature Neuroscience*, vol. 14, no. 2, pp. 139–142, 2011.

[4] G. B. Stanley, “Reading and writing the neural code,” *Nature Neuroscience*, vol. 16, pp. 259–263, 2013.

[5] Z. Chen, T. W. Berger, A. Cichocki, K. G. Oweiss, R. Quian Quiroga, and N. V. Thakor, “Signal processing for neural spike trains,” *Computational Intelligence and Neuroscience*, vol. 2010, Article ID 698751, 2 pages, 2010.

[6] J. Macke, P. Berens, and M. Bethge, “Statistical analysis of multicell recordings: linking population coding models to experimental data,” *Frontiers in Computational Neuroscience*, vol. 5, article 35, 2011.

[7] J. Bernardo and A. F. M. Smith, *Bayesian Theory*, John & Wiley, New York, NY, USA, 1994.

[8] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian Data Analysis*, Chapman & Hall/CRC, New York, NY, USA, 2nd edition, 2004.

[9] Y. Pawitan, *In All Likelihood: Statistical Modelling and Inference Using Likelihood*, Clarendon Press, New York, NY, USA, 2001.

[10] D. J. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes*, Springer, New York, NY, USA, 2nd edition, 2003.

[11] E. N. Brown, R. Barbieri, U. T. Eden, and L. M. Frank, “Likelihood methods for neural data analysis,” in *Computational Neuroscience: A Comprehensive Approach*, J. Feng, Ed., pp. 253–286, CRC Press, New York, NY, USA, 2003.

[12] E. N. Brown, “Theory of point processes for neural systems,” in *Methods and Models in Neurophysics*, C. C. Chow, B. Gutkin, D. Hansel et al., Eds., pp. 691–727, Elsevier, San Diego, Calif, USA, 2005.

[13] Z. Chen, R. Barbieri, and E. N. Brown, “State-space modeling of neural spike train and behavioral data,” in *Statistical Signal Processing for Neuroscience and Neurotechnology*, K. Oweiss, Ed., pp. 161–200, Elsevier, San Diego, Calif, USA, 2010.

[14] W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue, and E. N. Brown, “A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects,” *Journal of Neurophysiology*, vol. 93, no. 2, pp. 1074–1089, 2005.

[15] P. McCullagh and A. Nelder, *Generalized Linear Models*, vol. 22 of *Computational Intelligence and Neuroscience*, Chapman & Hall/CRC Press, New York, NY, USA, 2nd edition, 1989.

[16] E. Schneidman, M. J. Berry II, R. Segev, and W. Bialek, “Weak pairwise correlations imply strongly correlated network states in a neural population,” *Nature*, vol. 440, no. 7087, pp. 1007–1012, 2006.

[17] H. Nasser, O. Marre, and B. Cessac, “Spatio-temporal spike train analysis for large scale networks using the maximum entropy principle and Monte Carlo method,” *Journal of Statistical Mechanics*, vol. 2013, Article ID P03006, 2013.

[18] E. N. Brown, R. Barbieri, V. Ventura, R. E. Kass, and L. M. Frank, “The time-rescaling theorem and its application to neural spike train data analysis,” *Neural Computation*, vol. 14, no. 2, pp. 325–346, 2002.

[19] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, “A new method for the nonlinear transformation of means and covariances in filters and estimators,” *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 477–482, 2000.

[20] S. Särkkä, “On unscented Kalman filtering for state estimation of continuous-time nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 52, no. 9, pp. 1631–1641, 2007.

[21] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, “Introduction to variational methods for graphical models,” *Machine Learning*, vol. 37, no. 2, pp. 183–233, 1999.

[22] H. Attias, “A variational Bayesian framework for graphical models,” in *Advances in Neural Information Processing Systems (NIPS) 12*, S. A. Solla, T. K. Leen, and K. R. Müller, Eds., MIT Press, Boston, Mass, USA, 2000.

[23] M. Beal and Z. Ghahramani, "Variational Bayesian learning of directed graphical models," *Bayesian Analysis*, vol. 1, no. 4, pp. 793–832, 2006.

[24] D. J. MacKay, *Information Theory, Inference, and Learning Algorithms*, Cambridge University Press, New York, NY, USA, 2003.

[25] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, New York, NY, USA, 2006.

[26] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*, MIT Press, Cambridge, Mass, USA, 2012.

[27] D. Barber, *Bayesian Reasoning and Machine Learning*, Cambridge University Press, New York, NY, USA, 2012.

[28] D. Barber, A. T. Cemgil, and S. Chiappa, *Bayesian Time Series Models*, Cambridge University Press, New York, NY, USA, 2011.

[29] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, New York, NY, USA, 2nd edition, 2006.

[30] A. Dempster, N. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society B*, vol. 39, pp. 1–38, 1977.

[31] K. Katahira, K. Watanabe, and M. Okada, "Deterministic annealing variant of variational Bayes method," *Journal of Physics*, vol. 95, no. 1, Article ID 012015, 2008.

[32] K. Kurihara and M. Welling, "Bayesian *k*-means as a "maximization-expectation" algorithm," *Neural Computation*, vol. 21, no. 4, pp. 1145–1172, 2009.

[33] J. Sung, Z. Ghahramani, and S.-Y. Bang, "Latent-space variational bayes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 12, pp. 2236–2242, 2008.

[34] J. Sung, Z. Ghahramani, and S.-Y. Bang, "Second-order latent-space variational bayes for approximate bayesian inference," *IEEE Signal Processing Letters*, vol. 15, pp. 918–921, 2008.

[35] R. E. Turner and M. Sahani, "Two problems with variational expectation maximisation for time series models," in *Bayesian Time Series Models*, D. Barber, A. T. Cemgil, and S. Chiappa, Eds., pp. 115–138, Cambridge University Press, New York, NY, USA, 2011.

[36] K. Watanabe, "An alternative view of variational Bayes and asymptotic approximations of free energy," *Machine Learning*, vol. 86, no. 2, pp. 273–293, 2012.

[37] A. Honkela, T. Raiko, M. Kuusela, M. Tornio, and J. Karhunen, "Approximate riemannian conjugate gradient learning for fixed-form variational bayes," *Journal of Machine Learning Research*, vol. 11, pp. 3235–3268, 2010.

[38] T. P. Minka, *A family of algorithms for approximate Bayesian inference [Ph.D. thesis]*, Department of EECS, Massachusetts Institute of Technology, Cambridge, Mass, USA, 2001.

[39] S.-I. Amari and H. Nagaoka, *Methods of Information Geometry*, Oxford University Press, New York, NY, USA, 2007.

[40] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, *Markov Chain Monte Carlo in Practice*, Chapman & Hall/CRC, New York, NY, USA, 1995.

[41] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer, New York, NY, USA, 2nd edition, 2004.

[42] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[43] W. K. Hastings, "Monte carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.

[44] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, 1984.

[45] R. M. Neal, "Suppressing random walks in Markov chain Monte Carlo using ordered overrelaxation," Tech. Rep. 9508, University of Toronto; Department of Statistics, 1995.

[46] T. Marshall and G. Roberts, "An adaptive approach to Langevin MCMC," *Statistics and Computing*, vol. 22, no. 5, pp. 1041–1057, 2012.

[47] Y. Qi and T. P. Minka, "Hessian-based Markov chain Monte-Carlo algorithms," in *Proceedings of the 1st Cape Cod Workshop on Monte Carlo Methods*, Cape Cod, Mass, USA, September 2002.

[48] P. J. Green, "Reversible jump Markov chain monte carlo computation and Bayesian model determination," *Biometrika*, vol. 82, no. 4, pp. 711–732, 1995.

[49] R. E. Kass and A. E. Raftery, "Bayes factors," *Journal of the American Statistical Association*, vol. 90, no. 430, pp. 773–795, 1995.

[50] M. Lavine and M. J. Schervish, "Bayes factors: what they are and what they are not," *American Statistician*, vol. 53, no. 2, pp. 119–122, 1999.

[51] S. M. Lewis and A. E. Raftery, "Estimating Bayes factors via posterior simulation with the Laplace-Metropolis estimator," *Journal of the American Statistical Association*, vol. 92, no. 438, pp. 648–655, 1997.

[52] T. Toni and M. P. H. Stumpf, "Simulation-based model selection for dynamical systems in systems and population biology," *Bioinformatics*, vol. 26, no. 1, pp. 104–110, 2009.

[53] R. M. Neal, *Bayesian Learning for Neural Networks*, Springer, New York, NY, USA, 1996.

[54] J. A. Hoeting, D. Madigan, A. E. Raftery, and C. T. Volinsky, "Bayesian model averaging: a tutorial," *Statistical Science*, vol. 14, no. 4, pp. 382–417, 1999.

[55] A. E. Raftery, "Approximate Bayes factors and accounting for model uncertainty in generalised linear models," *Biometrika*, vol. 83, no. 2, pp. 251–266, 1996.

[56] Z. Chen and E. N. Brown, "State space model," *Scholarpedia*, vol. 8, no. 3, Article ID 30868, 2013.

[57] L. Paninski, Y. Ahmadian, D. G. Ferreira et al., "A new look at state-space models for neural data," *Journal of Computational Neuroscience*, vol. 29, no. 1-2, pp. 107–126, 2010.

[58] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, New York, NY, USA, 4th edition, 2002.

[59] C. P. Robert, T. Rydén, and D. M. Titterington, "Bayesian inference in hidden Markov models through the reversible jump Markov chain Monte Carlo method," *Journal of the Royal Statistical Society B*, vol. 62, no. 1, pp. 57–75, 2000.

[60] S. L. Scott, "Bayesian methods for hidden Markov models: recursive computing in the 21st century," *Journal of the American Statistical Association*, vol. 97, no. 457, pp. 337–351, 2002.

[61] Z. Ghahramani, "Learning dynamic Bayesian networks," in *Adaptive Processing of Sequences and Data Structures*, C. L. Giles and M. Gori, Eds., pp. 168–197, Springer, New York, NY, USA, 1998.

[62] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME*, vol. 82, pp. 35–45, 1960.

[63] W. Wu, Y. Gao, E. Bienenstock, J. P. Donoghue, and M. J. Black, "Bayesian population decoding of motor cortical activity using a Kalman filter," *Neural Computation*, vol. 18, no. 1, pp. 80–118, 2006.

[64] W. Wu, J. E. Kulkarni, N. G. Hatsopoulos, and L. Paninski, "Neural decoding of hand motion using a linear state-space model with hidden states," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 17, no. 4, pp. 370–378, 2009.

[65] E. N. Brown, L. M. Frank, D. Tang, M. C. Quirk, and M. A. Wilson, "A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells," *Journal of Neuroscience*, vol. 18, no. 18, pp. 7411–7425, 1998.

[66] A. C. Smith and E. N. Brown, "Estimating a state-space model from point process observations," *Neural Computation*, vol. 15, no. 5, pp. 965–991, 2003.

[67] U. T. Eden, L. M. Frank, R. Barbieri, V. Solo, and E. N. Brown, "Dynamic analysis of neural encoding by point process adaptive filtering," *Neural Computation*, vol. 16, no. 5, pp. 971–998, 2004.

[68] S. Koyama, L. Castellanos Pérez-Bolde, C. Rohilla Shalizi, and R. E. Kass, "Approximate methods for state-space models," *Journal of the American Statistical Association*, vol. 105, no. 489, pp. 170–180, 2010.

[69] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*, Springer, New York, NY, USA, 2001.

[70] A. E. Brockwell, A. L. Rojas, and R. E. Kass, "Recursive Bayesian decoding of motor cortical signals by particle filtering," *Journal of Neurophysiology*, vol. 91, no. 4, pp. 1899–1907, 2004.

[71] A. Ergun, R. Barbieri, U. T. Eden, M. A. Wilson, and E. N. Brown, "Construction of point process adaptive filter algorithms for neural system using sequential Monte Carlo methods," *IEEE Transactions on Biomedical Engineering*, vol. 54, pp. 419–428, 2007.

[72] V. Šmídl and A. Quinn, "Variational Bayesian filtering," *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 5020–5030, 2008.

[73] Y. Salimpour, H. Soltanian-Zadeh, S. Salehi, N. Emadi, and M. Abouzari, "Neuronal spike train analysis in likelihood space," *PLoS ONE*, vol. 6, no. 6, Article ID e21256, 2011.

[74] N. L. Hjort, C. Holmes, P. Müller, and S. G. Walker, *Bayesian Nonparametrics*, Cambridge University Press, New York, NY, USA, 2010.

[75] Z. Ghahramani, "Bayesian nonparametrics and the probabilistic approach to modeling," *Philosophical Transactions on Royal Society of London A*, vol. 371, Article ID 20110553, 2012.

[76] E. Fox, E. Sudderth, M. Jordan, and A. Willsky, "Bayesian nonparametric methods for learning markov switching processes," *IEEE Signal Processing Magazine*, vol. 27, no. 6, pp. 43–54, 2010.

[77] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, Mass, USA, 2005.

[78] J. Van Gael, Y. Saatci, Y. W. Teh, and Z. Ghahramani, "Beam sampling for the infinite hidden Markov model," in *25th International Conference on Machine Learning*, pp. 1088–1095, fin, July 2008.

[79] F. Gabbiani and C. Koch, "Principles of spike train analysis," in *Methods in Neuronal Modeling: From Synapses to Networks*, C. Koch and I. Segev, Eds., pp. 313–360, MIT Press, Boston, Mass, USA, 2nd edition, 1998.

[80] R. E. Kass, V. Ventura, and E. N. Brown, "Statistical issues in the analysis of neuronal data," *Journal of Neurophysiology*, vol. 94, no. 1, pp. 8–25, 2005.

[81] J. S. Prentice, J. Homann, K. D. Simmons, G. Tkačik, V. Balasubramanian, and P. C. Nelson, "Fast, scalable, bayesian spike identification for Multi-Electrode arrays," *PLoS ONE*, vol. 6, no. 7, Article ID e19884, 2011.

[82] F. Wood, M. J. Black, C. Vargas-Irwin, M. Fellows, and J. P. Donoghue, "On the variability of manual spike sorting," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 912–918, 2004.

[83] C. Ekanadham, D. Tranchina, and E. P. Simoncelli, "A blind deconvolution method for neural spike identification," in *Proceedings of the 25th Annual Conference on Neural Information Processing Systems (NIPS '11)*, vol. 23, MIT Press, December 2011.

[84] M. S. Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials," *Network*, vol. 9, no. 4, pp. R53–R78, 1998.

[85] D. P. Nguyen, L. M. Frank, and E. N. Brown, "An application of reversible-jump Markov chain Monte Carlo to spike classification of multi-unit extracellular recordings," *Network*, vol. 14, no. 1, pp. 61–82, 2003.

[86] F. Wood and M. J. Black, "A nonparametric Bayesian alternative to spike sorting," *Journal of Neuroscience Methods*, vol. 173, no. 1, pp. 1–12, 2008.

[87] J. A. Herbst, S. Gammeter, D. Ferrero, and R. H. R. Hahnloser, "Spike sorting with hidden Markov models," *Journal of Neuroscience Methods*, vol. 174, no. 1, pp. 126–134, 2008.

[88] A. Calabrese and L. Paninski, "Kalman filter mixture model for spike sorting of non-stationary data," *Journal of Neuroscience Methods*, vol. 196, no. 1, pp. 159–169, 2011.

[89] V. Ventura, "Automatic spike sorting using tuning information," *Neural Computation*, vol. 21, no. 9, pp. 2466–2501, 2009.

[90] V. Ventura, "Traditional waveform based spike sorting yields biased rate code estimates," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 106, no. 17, pp. 6921–6926, 2009.

[91] M. Park and J. W. Pillow, "Receptive field inference with localized priors," *PLoS Computational Biology*, vol. 7, no. 10, Article ID e1002219, 2011.

[92] I. M. Park and J. W. Pillow, "Bayesian spike-triggered covariance analysis," in *Advances in Neural Information Processing Systems (NIPS)*, J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Fereira, and K. Q. Weinberger, Eds., vol. 24, pp. 1692–1700, MIT Press, Boston, Mass, USA, 2011.

[93] D. Endres and M. Oram, "Feature extraction from spike trains with Bayesian binning: 'Latency is where the signal starts'," *Journal of Computational Neuroscience*, vol. 29, no. 1-2, pp. 149–169, 2010.

[94] I. Dimatteo, C. R. Genovese, and R. E. Kass, "Bayesian curve-fitting with free-knot splines," *Biometrika*, vol. 88, no. 4, pp. 1055–1071, 2001.

[95] A. C. Smith, J. D. Scalon, S. Wirth, M. Yanike, W. A. Suzuki, and E. N. Brown, "State-space algorithms for estimating spike rate functions," *Computational Intelligence and Neuroscience*, vol. 2010, Article ID 426539, 2010.

[96] B. Cronin, I. H. Stevenson, M. Sur, and K. P. Körding, "Hierarchical bayesian modeling and Markov chain Monte Carlo sampling for tuning-curve analysis," *Journal of Neurophysiology*, vol. 103, no. 1, pp. 591–602, 2010.

[97] H. Taubman, E. Vaadia, R. Paz, and G. Chechik, "A Bayesian approach for characterizing direction tuning curves in the supplementary motor area of behaving monkeys," *Journal of Neurophysiology*, 2013.

[98] L. Paninski, J. Pillow, and J. Lewi, "Statistical models for neural encoding, decoding, and optimal stimulus design," in *Computational Neuroscience: Theoretical Insights Into Brain Function*, P. Cisek, T. Drew, and J. Kalaska, Eds., Elsevier, 2007.

[99] S. Gerwinn, J. H. Macke, M. Seeger, and M. Bethge, "Bayesian inference for spiking neuron models with a sparsity prior," in *Advances in Neural Information Processing Systems (NIPS)*, J. C. Platt, D. Koller, Y. Singer, and S. Roweis, Eds., vol. 20, pp. 529–536, MIT Press, Boston, Mass, USA, 2008.

[100] J. W. Pillow and J. G. Scott, "Fully Bayesian inference for neural models with negative-binomial spiking," in *Advances in Neural Information Processing Systems (NIPS)*, P. Bartlett, F. C. N. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., vol. 25, pp. 1907–1915, MIT Press, Boston, Mass, USA, 2012.

[101] S. Koyama, U. T. Eden, E. N. Brown, and R. E. Kass, "Bayesian decoding of neural spike trains," *Annals of the Institute of Statistical Mathematics*, vol. 62, no. 1, pp. 37–59, 2010.

[102] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, New York, NY, USA, 2004.

[103] J. W. Pillow, Y. Ahmadian, and L. Paninski, "Model-based decoding, information estimation, and change-point detection techniques for multineuron spike trains," *Neural Computation*, vol. 23, no. 1, pp. 1–45, 2011.

[104] A. D. Ramirez, Y. Ahmadian, J. Schumacher, D. Schneider, S. M. N. Woolley, and L. Paninski, "Incorporating naturalistic correlation structure improves spectrogram reconstruction from neuronal activity in the songbird auditory midbrain," *Journal of Neuroscience*, vol. 31, no. 10, pp. 3828–3842, 2011.

[105] Z. Chen, K. Takahashi, and N. G. Hatsopoulos, "Sparse Bayesian inference methods for decoding 3D reach and grasp kinematics and joint angles with primary motor cortical ensembles," in *Proceedings of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology (EMBC '13)*, pp. 5930–5933, 2013.

[106] K. Zhang, I. Ginzburg, B. L. McNaughton, and T. J. Sejnowski, "Interpreting neuronal population activity by reconstruction: unified framework with application to hippocampal place cells," *Journal of Neurophysiology*, vol. 79, no. 2, pp. 1017–1044, 1998.

[107] W. Truccolo, G. M. Friehs, J. P. Donoghue, and L. R. Hochberg, "Primary motor cortex tuning to intended movement kinematics in humans with tetraplegia," *Journal of Neuroscience*, vol. 28, no. 5, pp. 1163–1178, 2008.

[108] W. Truccolo and J. P. Donoghue, "Nonparametric modeling of neural point processes via stochastic gradient boosting regression," *Neural Computation*, vol. 19, no. 3, pp. 672–705, 2007.

[109] T. P. Coleman and S. S. Sarma, "A computationally efficient method for nonparametric modeling of neural spiking activity with point processes," *Neural Computation*, vol. 22, no. 8, pp. 2002–2030, 2010.

[110] M. M. Shanechi, E. N. Brown, and Z. M. Williams, "Neural population partitioning and a concurrent brain-machine interface for sequential control motor function," *Nature Neuroscience*, vol. 12, pp. 1715–1722, 2012.

[111] M. M. Shanechi, G. W. Wornell, Z. Williams, and E. N. Brown, "A parallel point-process filter for estimation of goal-directed movements from neural signals," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '10)*, pp. 521–524, Dallas, Tex, USA, March 2010.

[112] Y. Ahmadian, J. W. Pillow, and L. Paninski, "Efficient Markov chain monte carlo methods for decoding neural spike trains," *Neural Computation*, vol. 23, no. 1, pp. 46–96, 2011.

[113] A. K. Bansal, W. Truccolo, C. E. Vargas-Irwin, and J. P. Donoghue, "Decoding 3D reach and grasp from hybrid signals in motor and premotor cortices: spikes, multiunit activity, and local field potentials," *Journal of Neurophysiology*, vol. 107, no. 5, pp. 1337–1355, 2012.

[114] V. Ventura, "Spike train decoding without spike sorting," *Neural Computation*, vol. 20, no. 4, pp. 923–963, 2008.

[115] Z. Chen, F. Kloosterman, S. Layton, and W. A. Wilson, "Transductive neural decoding of unsorted neuronal spikes of rat hippocampus," in *Proceedings of the 34th Annual International Conference of the IEEE Engineering in Medicine and Biology (EMBC '12)*, pp. 1310–1313, August 2012.

[116] F. Kloosterman, S. Layton, Z. Chen, and M. A. Wilson, "Bayesian decoding of unsorted spikes in the rat hippocampus," *Journal of Neurophysiology*, 2013.

[117] A. L. Jacobs, G. Fridman, R. M. Douglas et al., "Ruling out and ruling in neural codes," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 106, no. 14, pp. 5936–5941, 2009.

[118] D. H. Johnson, "Information theory and neural information processing," *IEEE Transactions on Information Theory*, vol. 56, no. 2, pp. 653–666, 2010.

[119] C. Smith and L. Paninski, "Computing loss of efficiency in optimal Bayesian decoders given noisy or incomplete spike trains," *Network*, vol. 24, no. 2, pp. 75–98, 2013.

[120] D. S. Greenberg, A. R. Houweling, and J. N. D. Kerr, "Population imaging of ongoing neuronal activity in the visual cortex of awake rats," *Nature Neuroscience*, vol. 11, no. 7, pp. 749–751, 2008.

[121] J. T. Vogelstein, B. O. Watson, A. M. Packer, R. Yuste, B. Jedynak, and L. Paninskik, "Spike inference from calcium imaging using sequential Monte Carlo methods," *Biophysical Journal*, vol. 97, no. 2, pp. 636–655, 2009.

[122] J. T. Vogelstein, A. M. Packer, T. A. Machado et al., "Fast nonnegative deconvolution for spike train inference from population calcium imaging," *Journal of Neurophysiology*, vol. 104, no. 6, pp. 3691–3704, 2010.

[123] C. Andrieu, E. Barat, and A. Doucet, "Bayesian deconvolution of noisy filtered point processes," *IEEE Transactions on Signal Processing*, vol. 49, no. 1, pp. 134–146, 2001.

[124] J. Oñativia, S. R. Schultz, and P. L. Dragotti, "A finite rate of innovation algorithm for fast and accurate spike detection from two-photon calcium imaging," *Journal of Neural Engineering*, vol. 10, Article ID 046017, 2013.

[125] J. W. Pillow, J. Shlens, L. Paninski et al., "Spatio-temporal correlations and visual signalling in a complete neuronal population," *Nature*, vol. 454, no. 7207, pp. 995–999, 2008.

[126] W. Truccolo, L. R. Hochberg, and J. P. Donoghue, "Collective dynamics in human and monkey sensorimotor cortex: predicting single neuron spikes," *Nature Neuroscience*, vol. 13, no. 1, pp. 105–111, 2010.

[127] E. S. Chornoboy, L. P. Schramm, and A. F. Karr, "Maximum likelihood identification of neural point process systems," *Biological Cybernetics*, vol. 59, no. 4-5, pp. 265–275, 1988.

[128] M. Okatan, M. A. Wilson, and E. N. Brown, "Analyzing functional connectivity using a network likelihood model of ensemble neural spiking activity," *Neural Computation*, vol. 17, no. 9, pp. 1927–1961, 2005.

[129] F. Rigat, M. de Gunst, and J. van Pelt, "Bayesian modelling and analysis of spatio-temporal neuronal networks," *Bayesian Analysis*, vol. 1, no. 4, pp. 733–764, 2006.

[130] I. H. Stevenson, J. M. Rebesco, N. G. Hatsopoulos, Z. Haga, L. E. Miller, and K. P. Kording, "Bayesian inference of functional connectivity and network structure from spikes," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 17, no. 3, pp. 203–213, 2009.

[131] Z. Chen, D. F. Putrino, S. Ghosh, R. Barbieri, and E. N. Brown, "Statistical inference for assessing functional connectivity of neuronal ensembles with sparse spiking data," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 2, pp. 121–135, 2011.

[132] S. Eldawlatly, Y. Zhou, R. Jin, and K. G. Oweiss, "On the use of dynamic Bayesian networks in reconstructing functional neuronal networks from spike train ensembles," *Neural Computation*, vol. 22, no. 1, pp. 158–189, 2010.

[133] Y. Mishchenko, J. Vogelstein, and L. Paninski, "A Bayesian approach for inferring neuronal connectivity from calcium uorescent imaging data," *Annals of Applied Statistics*, vol. 5, pp. 1229–1261, 2011.

[134] L. Martignon, G. Deco, K. Laskey, M. Diamond, W. Freiwald, and E. Vaadia, "Neural coding: higher-order temporal patterns in the neurostatistics of cell assemblies," *Neural Computation*, vol. 12, no. 11, pp. 2621–2653, 2000.

[135] H. Shimazaki, S. Amari, E. N. Brown, and S. Gruen, "State-space analysis of time-varying higherorder spike correlation for multiple neural spike train data," *PLoS Computational Biology*, vol. 8, no. 3, Article ID e1002385, 2012.

[136] B. M. Turner, B. U. Forstmann, E.-J. Wagenmakers, S. D. Brown, P. B. Sederberg, and M. Steyvers, "A Bayesian framework for simultaneously modeling neural and behavioral data," *NeuroImage*, vol. 72, pp. 193–206, 2013.

[137] J. W. Pillow and P. Latham, "Neural characterization in partially observed populations of spiking neurons," in *Advances in Neural Information Processing Systems (NIPS)*, J. C. Platt, D. Koller, Y. Singer, and S. Roweis, Eds., vol. 20, pp. 1161–1168, MIT Press, Boston, Mass, USA, 2008.

[138] L. Li, I. M. Park, S. Seth, J. C. Sanchez, and J. C. Príncipe, "Functional connectivity dynamics among cortical neurons: a dependence analysis," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 20, no. 1, pp. 18–30, 2012.

[139] R. E. Kass, R. C. Kelly, and W.-L. Loh, "Assessment of synchrony in multiple neural spike trains using loglinear point process models," *The Annals of Applied Statistics*, vol. 5, no. 2B, pp. 1262–1292, 2011.

[140] S. Kim, D. Putrino, S. Ghosh, and E. N. Brown, "A Granger causality measure for point process models of ensemble neural spiking activity," *PLoS Computational Biology*, vol. 7, no. 3, Article ID e1001110, 2011.

[141] R. Vicente, M. Wibral, M. Lindner, and G. Pipa, "Transfer entropy-a model-free measure of effective connectivity for the neurosciences," *Journal of Computational Neuroscience*, vol. 30, no. 1, pp. 45–67, 2011.

[142] P. Berkes, F. Wood, and J. Pillow, "Characterizing neural dependencies with copula models," in *Advances in Neural Information Processing Systems (NIPS)*, J. C. Platt, D. Koller, Y. Singer, and S. Roweis, Eds., vol. 20, MIT Press, Boston, Mass, USA, 2008.

[143] M. S. Smith, "Bayesian approaches to copula modelling," in *Bayesian Theory and Applications*, P. Damien, P. Dellaportas, N. Polson, and D. Stephens, Eds., Oxford University Press, New York, NY, USA, 2013.

[144] B. M. Yu, J. P. Cunningham, G. Santhanam, S. I. Ryu, K. V. Shenoy, and M. Sahani, "Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity," *Journal of Neurophysiology*, vol. 102, no. 1, pp. 614–635, 2009.

[145] Z. Chen, F. Kloosterman, E. N. Brown, and M. A. Wilson, "Uncovering spatial topology represented by rat hippocampal population neuronal codes," *Journal of Computational Neuroscience*, vol. 33, no. 2, pp. 227–255, 2012.

[146] Z. Chen, S. N. Gomperts, J. Yamamoto, and W. A. Wilson, "Neural representation of spatial topology in the rodent hippocampus," *Neural Computation*, vol. 26, no. 1, pp. 1–39, 2014.

[147] Z. Chen and M. A. Wilson, "A variational nonparametric Bayesian approach for inferring rat hippocampal population codes," in *Proceedings of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology (EMBC '13)*, pp. 7092–7095, 2013.

[148] K. Famm, B. Litt, K. J. Tracey, E. S. Boyden, and M. Slaoui, "A jump-start for electroceuticals," *Nature*, vol. 496, pp. 159–161, 2013.