Biophysical Review

A Primer on Bayesian Inference for Biophysical Systems

Keegan E. Hines[1,*]
[1]Department of Neuroscience, University of Texas at Austin, Austin, Texas

ABSTRACT Bayesian inference is a powerful statistical paradigm that has gained popularity in many fields of science, but adoption has been somewhat slower in biophysics. Here, I provide an accessible tutorial on the use of Bayesian methods by focusing on example applications that will be familiar to biophysicists. I first discuss the goals of Bayesian inference and show simple examples of posterior inference using conjugate priors. I then describe Markov chain Monte Carlo sampling and, in particular, discuss Gibbs sampling and Metropolis random walk algorithms with reference to detailed examples. These Bayesian methods (with the aid of Markov chain Monte Carlo sampling) provide a generalizable way of rigorously addressing parameter inference and identifiability for arbitrarily complicated models.

The proper analysis and interpretation of experimental data are vital in the endeavor to understand natural phenomena. Here I describe the use of Bayesian inference, a statistical paradigm that has gained popularity in many fields including astrophysics (14), systems biology (12), and econometrics (6), among others. However, the adoption of Bayesian methods has been relatively slower in the study of protein biophysics, a field that relies primarily on more classical techniques. It is not my intention here to argue the merits of Bayesian methods over others, as this has been discussed previously (1,11,24). Instead, my aim is to provide an accessible introduction and tutorial on the use of these methods with a focus on problems that should be familiar to the experimental biophysicist.

Bayesian inference

Suppose that we make some measurements, y, and want to use these data to gain an accurate estimate of some model parameters, θ. In Bayesian inference, the primary goal is to compute the posterior distribution. This is a probability distribution over the parameter space that quantifies how probable it is that a particular value of the parameter(s) has given rise to the observed data. This distribution provides not only an optimal point estimate of the parameters (the maximum a posteriori or MAP estimate), but also a quantification of the entire parameter space, yielding a straightforward way to calculate confidence intervals. In this way, we consider the entire parameter space and ask which regions are most probable, given the data we saw. In some special cases, we can derive simple expressions for posterior distributions by using conjugate models. For more complex models, we can take advantage of computational methods that allow us to estimate posterior distributions of arbitrarily high dimension.

Consider that we treat not only the data y as random, but also treat the parameters of interest θ as random variables. We then need to address the joint probability of all random variables, $p(y,\theta)$. From the definition of conditional probability, we can write

$$p(y,\theta) = p(y|\theta)p(\theta) = p(\theta|y)p(y). \quad (1)$$

We have two expressions for the joint density $p(y,\theta)$, and we can equate them and rearrange to yield Bayes' rule,

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)}. \quad (2)$$

By treating both the parameters and the data as random variables, a simple manipulation of conditional probabilities yields a general expression for $p(\theta|y)$, the posterior distribution of the parameters. The other components of Bayes' rule are: $p(y|\theta)$, the likelihood of seeing the data given the parameters; $p(\theta)$, the prior distribution of the parameters; and $p(y)$, the marginal likelihood of the data. In practice, we generally only need to quantify the posterior distribution up to a constant of proportionality, so $p(y)$ is often ignored because it is independent of θ. This yields a more common form of Bayes' rule,

$$p(\theta|y) \propto p(y|\theta)p(\theta). \quad (3)$$

Computing the posterior distribution is then simply a matter of deciding upon the likelihood and the prior distribution and combining them. I will next show that if we put a little thought into finding prior distributions that are conjugate to the likelihood, then we can arrive at a simple expression for the posterior. Because we will not always be able to use a conjugate prior, I will later discuss the powerful

Submitted November 18, 2014, and accepted for publication March 18, 2015.

*Correspondence: keegan.hines@utexas.edu

Editor: Chris Lingle.

0006-3495/15/05/2103/11 $2.00

http://dx.doi.org/10.1016/j.bpj.2015.03.042

computational methods that allow to us calculate arbitrarily complicated posterior distributions.

Conjugate models

I will motivate our first foray into Bayesian modeling by taking as an example the experimental method of single molecule photobleaching (9,28). This is a powerful method for determining the interaction and stoichiometries of protein complexes. The strategy consists of tagging a fluorescent probe to a protein subunit of interest and then imaging single molecules. After sufficient time, the fluorophores will photobleach, and by counting the number of photobleaching events, we get a direct readout of how many subunits are associated. However, there is a nonnegligible probability that a fluorophore is already bleached before the measurement started. We will quantify this probability of being prebleached as $1-\theta$; that is, θ is the probability that a fluorophore bleaching event will be successfully detected. The result of this prebleaching is that a complex of n molecules might result in less than n bleaching events. Therefore, the ensemble of many such counts will be binomially distributed such that the probability of seeing y bleaching steps when n are possible goes as

$$p(y|\theta) = \frac{n!}{(n-y)!y!}\theta^{y}(1-\theta)^{n-y}. \tag{4}$$

As a simple inference problem, let us suppose that we want to estimate the prebleaching probability of an unknown fluorophore. To do this, we use a protein system that is well known so that we can assert that n is fixed to some known value. We perform a photobleaching experiment and gather N independent observations of bleaching counts and denote the total dataset as y_N. Our goal is then to estimate θ from y_N. Because each data point is drawn independently, Bayes' rule is

$$p(\theta|y_N) \propto \left(\prod_{i=1}^{N} p(y_i|\theta)\right) p(\theta). \tag{5}$$

Because we know the likelihood is a binomial distribution, we can begin to fill in the components of Bayes' rule:

$$p(\theta|y_N) \propto \left(\prod_{i=1}^{N} \frac{n!}{(n-y_i)!y_i!}\theta^{y_i}(1-\theta)^{n-y_i}\right) p(\theta). \tag{6}$$

All that remains is to decide on a form of the prior distribution over θ. Because θ is the probability of a binary event, it will be useful to utilize a distribution that is defined only on the unit interval. More importantly, it will be very useful if we choose a prior distribution that combines with a binomial likelihood in a helpful way. A distribution that accomplishes both of these goals is the beta distribution, Be(θ;a,b),

$$\mathrm{Be}(\theta;a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\theta^{a-1}(1-\theta)^{b-1}, \tag{7}$$

where $\Gamma(\,)$ is the gamma function and the notation Be(θ;a,b) denotes that Be( ) is a probability distribution over variable θ and is parameterized by a and b. Depending on how we choose the hyperparameters a and b, we can quantify any prior confidence we have about the value of θ. Alternatively, letting $a=b=1$ results in a flat prior distribution over θ. Fig. 1 shows beta distributions of different values of a and b. Note that this distribution provides a very flexible way for us to quantify any prior knowledge we might have, or we can adopt a flat prior. Therefore, our choice of prior distribution, which is often motivated by mathematical convenience, does not necessarily introduce systematic bias in our parameter estimates.

The most useful outcome of using a beta prior is that this distribution is conjugate to our binomial likelihood. Returning to Bayes' rule, we now have a form for both the likelihood and the prior in our model:

$$\begin{aligned} p(\theta|y_N) \propto &\left(\prod_{i=1}^{N} \frac{n!}{(n-y_i)!y_i!}\theta^{y_i}(1-\theta)^{n-y_i}\right) \\ &\times \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\theta^{a-1}(1-\theta)^{b-1}. \end{aligned} \tag{8}$$

We can remove some terms that do not depend on θ and we still retain a distribution that is proportional to the posterior distribution,

$$p(\theta|y_N) \propto \theta^{a-1}(1-\theta)^{b-1}\prod_{i=1}^{N}\theta^{y_i}(1-\theta)^{n-y_i}. \tag{9}$$

It is now obvious that we can combine the components from the likelihood and the prior,

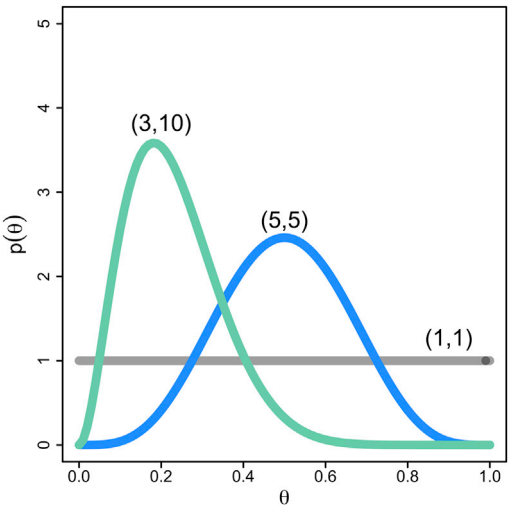

FIGURE 1 The beta distribution is shown with three different parameterizations. Used as a prior, the beta distribution provides a flexible way to quantify any prior knowledge we might have, or to specify a lack of prior knowledge. To see this figure in color, go online.

$$p(\theta|y_N) \propto \theta^{a-1}(1-\theta)^{b-1}\theta^{\left(\sum_{i=1}^{N} y_i\right)}(1-\theta)^{\left(\sum_{i=1}^{N} n-y_i\right)} \quad (10)$$

$$= \theta^{\left(\sum_{i=1}^{N} y_i\right)+a-1}(1-\theta)^{\left(\sum_{i=1}^{N} n-y_i\right)+b-1}. \quad (11)$$

Note that this form of the posterior distribution has the same basic form as a β-distribution. That is, the posterior distribution of θ is

$$p(\theta|y_N) = \mathrm{Be}(\theta; A, B), \quad (12)$$

where

$$A = a + \sum_{i=1}^{N} y_i, \quad (13)$$

$$B = b + \sum_{i=1}^{N} n - y_i. \quad (14)$$

This is the primary benefit of thinking carefully about our prior distribution. A conjugate prior combines naturally with the likelihood and results in a posterior distribution of the same functional form as the prior. Therefore, the posterior will have a simple closed form with parameters that are easily calculated from the data.

This example problem is continued in Fig. 2. In the left column are two simulated datasets drawn from binomial distributions with $n = 4$ and θ equal to 0.8 (*top*) and 0.5 (*bottom*). The right column shows the corresponding posterior distributions for θ. In this example, the hyperparameters of the prior distribution were both set to 1, which resulted in a flat prior distribution. Because of this, the peak of the posterior (MAP estimate) corresponds exactly to what we would estimate by maximizing the likelihood (i.e., finding the best fit to the data). In addition to this point estimate, we also have a quantification of the whole parameter space and would easily be able to quantify parameter confidence and construct confidence intervals. Therefore, by choosing a conjugate prior, calculating the full posterior distribution over the parameters is achieved effortlessly.

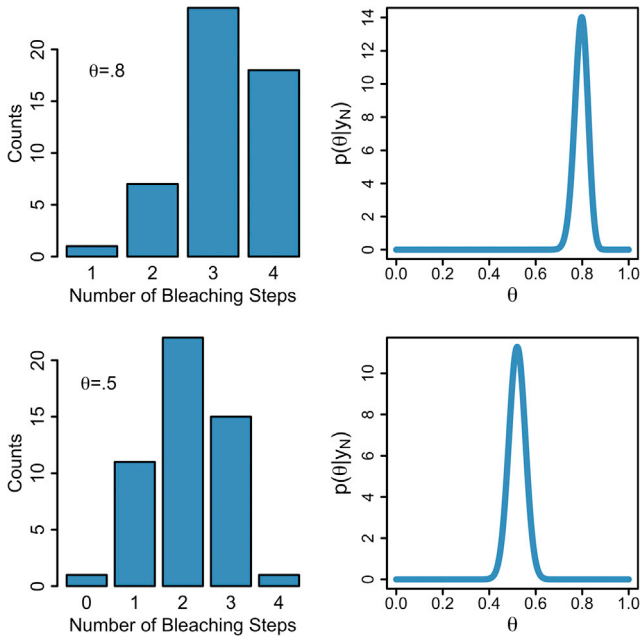

FIGURE 2 Posterior estimation in the beta-binomial model. (*Left*) Samples drawn from a binomial distribution with $n = 4$ and $\theta = 0.8$ (*top*) and $\theta = 0.6$ (*bottom*). A total of $N = 50$ samples are drawn. (*Right*) The resulting posterior distributions of θ. To see this figure in color, go online.

I will describe one more example of a conjugate model that will also serve to transition us toward more generally applicable computational methods. Imagine that we have used patch-clamp recording in order to measure the currents through a single ion channel. The transitions between open and closed states should follow Markovian dynamics, which prescribes that the duration of time spent in any state should be exponentially distributed. From our single channel recording, we tabulate the durations of each dwell-time and are left with a set of exponentially distributed random variables. It is our goal to estimate the corresponding timescale parameters of each distribution. Previous authors have thoroughly established successful methods for calculating these parameters using maximum likelihood methods (2), but I describe the Bayesian way of approaching this problem.

Again, we imagine the data are drawn from a single exponential distribution with unknown timescale parameter,

$$y_i \sim \theta e^{(-\theta y)}, \quad (15)$$

where I have introduced the notation $y \sim f(\,)$ to denote that the random variable y is sampled from the distribution $f(\,)$. Given some data y_N, we want to estimate the posterior distribution over θ. Recalling Bayes' rule,

$$p(\theta|y_N) \propto \left(\prod_{i=1}^{N} p(y_i|\theta)\right) p(\theta) \quad (16)$$

$$= \left(\prod_{i=1}^{N} \theta e^{(-\theta y_i)}\right) p(\theta). \quad (17)$$

Again, we want to carefully choose $p(\theta)$ so that it combines usefully with $p(y_i|\theta)$. The conjugate distribution to an exponential likelihood is the gamma distribution (Ga),

$$\mathrm{Ga}(\theta; a, b) = \frac{b^a}{\Gamma(a)}\theta^{a-1}e^{(-\theta b)}. \quad (18)$$

Combining likelihood and prior, we arrive at

$$p(\theta|y_N) \propto \left(\prod_{i=1}^{N} p(y_i|\theta)\right) p(\theta) \tag{19}$$

$$= \left(\prod_{i=1}^{N} \theta e^{(-\theta y_i)}\right) \frac{b^a}{\Gamma(a)} \theta^{a-1} e^{(-\theta b)} \tag{20}$$

$$\propto \left(\prod_{i=1}^{N} \theta e^{(-\theta y_i)}\right) \theta^{a-1} e^{(-\theta b)} \tag{21}$$

$$= \theta^{a+N-1} e^{-\theta\left(b+\sum_{i=1}^{N} y_i\right)}. \tag{22}$$

We see that the posterior distribution of θ is a gamma distribution with parameters that can be calculated from the data,

$$p(\theta|y_N) = \mathrm{Ga}(\theta; A, B), \tag{23}$$

where

$$A = a + N, \tag{24}$$

$$B = b + \sum_{i=1}^{N} y_i. \tag{25}$$

I now extend this model into a more interesting case that will lead into our first computational method, Gibbs sampling. Instead of modeling the data as drawn from a single exponential distribution, consider that the data are drawn from a mixture of multiple exponential distributions, a common case for single ion channel recordings. We imagine that each component has a distinct timescale parameter θ and mixture weight w. The data are drawn from some number of distinct components as

$$y_i \sim w_1\theta_1 e^{-\theta_1 y} + w_2\theta_2 e^{-\theta_2 y} + \ldots + w_K\theta_K e^{-\theta_K y} \tag{26}$$

$$= \sum_{j=1}^{K} w_j\theta_j e^{-\theta_j y}, \tag{27}$$

where the mixture weights sum to 1,

$$\sum_{j=1}^{K} w_j = 1.$$

Without loss of generality, I focus on just a two-component exponential mixture for simplicity:

$$y_i \sim w_1\theta_1 e^{-\theta_1 y} + w_2\theta_2 e^{-\theta_2 y}. \tag{28}$$

In this two-component model, we know that $w_1 + w_2 = 1$ and could reparameterize these to a single free parameter, but I will continue with this notation in order to generalize for later results. Our task then becomes estimating from the data the four resulting free parameters (although truly only three free parameters as $w_1 + w_2 = 1$):

$$p(\theta_1, \theta_2, w_1, w_2|y_N) \propto \left(\prod_{i=1}^{N} p(y_i|\theta_1, \theta_2, w_1, w_2)\right) \times p(\theta_1, \theta_2, w_1, w_2). \tag{29}$$

We wish to estimate a four-dimensional posterior distribution that spans the parameter space of the two timescale parameters and the two weight parameters. This kind of model will likely not have a simple closed form for the posterior, no matter how clever we may try to be with conjugate priors. However, we will be able to estimate this distribution using a numerical method called Markov chain Monte Carlo sampling (MCMC). The general strategy with MCMC is that while we may not be able to express a simple form for the posterior distribution, we could approximate its properties if we can draw a large number of samples from the distribution in which we are interested. Importantly, even though we do not know the posterior distribution, we can draw samples by constructing a Markov chain whose limiting distribution is the desired target distribution. Then, by simulating this chain for many iterations, we draw many samples from the underlying distribution. Generating a Markov chain with a desired limiting distribution can be achieved in several ways, and I first describe Gibbs sampling.

Gibbs sampling

While we may not be able to devise a simple form for the posterior, $p(\theta_1,\theta_2,w_1,w_2|y_N)$, we can, with some care, devise a simple form for the conditional posterior of each parameter. As it turns out, this simple advance allows us to estimate the full posterior distribution using an MCMC algorithm called Gibbs sampling (4,5). Before returning to our example, I describe Gibbs sampling in general.

Consider a general joint probability distribution between two random variables, $p(A,B)$. From the definition of conditional probability,

$$p(A|B) = \frac{p(A,B)}{p(B)}, \tag{30}$$

$$p(A,B) = p(B)p(A|B), \tag{31}$$

$$p(A,B) \propto p(A|B). \tag{32}$$

Similarly, we could calculate the conditional density with respect to the other variable,

$$p(B|A) = \frac{p(A,B)}{p(A)}, \tag{33}$$

$$p(A,B) = p(A)p(B|A), \tag{34}$$

$$p(A,B) \propto p(B|A). \tag{35}$$

Thus, the joint distribution, $p(A,B)$, is linearly proportional to both conditional distributions, $p(A|B)$ and $p(B|A)$. This fact holds generally for joint distributions over any number of random variables and is the basis of Gibbs sampling. The strategy is that while the joint distribution, $p(A,B)$, might have no simple closed form, we can likely derive a simple form of each univariate conditional distribution. More concretely, suppose we have gathered some data y_N and want to estimate parameters A and B. The joint posterior $p(A,B|y_N)$ may have no simple, closed form. However, for any particular values a,b of the random variables A and B, the joint $p(A,B|y_N)$ is proportional to each of its univariate conditionals: $p(A,B|y_N) \propto p(A|y_N, B = b)$ and $p(A,B|y_N) \propto p(B|y_N, A = a)$. If we can find a simple form for each univariate conditional $p(A|y_N,B)$ and $p(B|y_N,A)$, then we can approximate $p(A,B|y_N)$ with a Markov chain that alternately samples $p(A|y_N,B)$ and $p(B|y_N,A)$. More generally, let $p(\theta_1,\ldots,\theta_K|x)$ be a K-dimensional posterior distribution with no simple closed form. If each univariate conditional distribution has a closed form such as $p(\theta_1|\theta_2,\ldots\ \theta_K,x) \propto F(\theta_1)$, then Gibbs sampling proceeds by sequentially sampling each parameter conditioned on the previous samples of all other parameters. For each iteration of the algorithm, we draw the ith random sample of each parameter according to the univariate conditional distributions,

$$\theta_1^i \sim p\left(\theta_1 \middle| \theta_2^{i-1}, \theta_3^{i-1}, \ldots, \theta_K^{i-1}, x\right) = F(\theta_1), \tag{36}$$

$$\theta_2^i \sim p\left(\theta_2 \middle| \theta_1^{i}, \theta_3^{i-1}, \ldots, \theta_K^{i-1}, x\right) = F(\theta_2), \tag{37}$$

$$\theta_3^i \sim p\left(\theta_3 \middle| \theta_1^{i}, \theta_2^{i}, \ldots, \theta_K^{i-1}, x\right) = F(\theta_3), \tag{38}$$

$$\ldots \tag{39}$$

$$\theta_K^i \sim p\left(\theta_K \middle| \theta_1^{i}, \theta_2^{i}, \ldots, \theta_{K-1}^{i}, x\right) = F(\theta_K). \tag{40}$$

Therefore, being able to draw samples from each univariate conditional posterior allows us to construct a K-dimensional Markov chain that explores the parameter space in proportion to the posterior probability.

I now return to the two-component exponential mixture model. Recall that we would be unable to devise a simple form for the four-dimensional posterior distribution, $p(\theta_1,\theta_2,w_1,w_2|y_N)$. However, we will see that it is straightforward to compute each conditional posterior, $p(\theta_1|\theta_2,w_1,w_2,y_N)$, and so on (for brevity, I now adopt the notation $p(\theta_1|\ldots)$ to denote a conditional probability with respect to all other random variables in the model).

First, I employ a trick known as data augmentation (a name that is somewhat of a misnomer, because we will be augmenting the parameters and not the data) by which we make the model more complicated in order to simplify the sampling scheme. In particular, I add new latent indicator variables $s_1,s_2,\ldots,s_N$ (one for each data point) that serve to label from which component a particular data point was likely drawn. For our two-component mixture model, each indicator variable points to one of the two mixture components, $s_i \in \{1,2\}$. Our posterior distribution now has many parameters,

$$p(\theta_1,\theta_2,w_1,w_2,s_1,\ldots,s_N|y_N) \propto \left(\prod_{i=1}^{N} p(y_i|\ldots)\right) \times p(\theta_1,\theta_2,w_1,w_2,s_1,\ldots,s_N), \tag{41}$$

but in the process of MCMC sampling, we marginalize out the latent variables s_i that we introduced,

$$p(\theta_1,\theta_2,w_1,w_2|y_N) = \int p(\theta_1,\theta_2,w_1,w_2,s_1,\ldots,s_N|y_N) \times ds_1 ds_2 \ldots ds_N, \tag{42}$$

$$p(\theta_1,\theta_2,w_1,w_2|y_N) = \sum_{\substack{s_i \in \{1,2\} \\ i=1..N}} p(\theta_1,\theta_2,w_1,w_2,s_1,\ldots,s_N|y_N). \tag{43}$$

Therefore, even though we made the model more complicated by adding the s_i, we return to the desired model when we marginalize out the latent variables, which will be achieved with MCMC sampling of those parameters.

To create our Gibbs sampler, we need the conditional posterior distribution of each parameter, which is composed only of those components from the likelihood and prior that are relevant to each parameter. We seek

$$p\left(\theta_j \middle| \ldots\right) \propto p(y_N|\ldots)p\left(\theta_j\right), \tag{44}$$

$$p\left(w_j \middle| \ldots\right) \propto p(y_N|\ldots)p\left(w_j\right), \tag{45}$$

$$p(s_i|\ldots) \propto p(y_i|\ldots)p(s_i). \tag{46}$$

Relying on our previous results, we simply need to devise a conjugate prior for each parameter and we will be able to sample from the corresponding conditional posterior. Now that we have the latent indicators s_i, let A_j be the set of all i such that $s_i = j$. For each component, j, we already know a good conjugate model for estimating θ_j: the exponential-Gamma model. Thus,

$$p\left(\theta_j \middle| \ldots\right) \propto \left(\prod_{i \in A_j} w_j e^{-\theta_j y_i}\right) \mathrm{Ga}\left(\theta_j; a, b\right) \tag{47}$$

$$= \mathrm{Ga}\left(\theta_j; a + |A_j|, b + \sum_{i \in A_j} y_i\right), \quad (48)$$

where $|A_j|$ denotes the number of elements in the set A_j. For each indicator variable, we need to sample s_i from the components {1,2} with probability equal to the posterior probability that data point i was drawn from each component. If we assume a flat prior on s_i, then this calculation boils down to calculating the likelihood that data point i was drawn from each component,

$$p(s_i = 1|\ldots) \propto p(y_i|s_i = 1, \ldots)p(s_i = 1), \quad (49)$$

$$\propto p(y_i|s_i = 1, \ldots), \quad (50)$$

$$\propto \theta_1 e^{-\theta_1 y_i}, \quad (51)$$

and

$$p(s_i = 2|\ldots) \propto p(y_i|s_i = 2, \ldots)p(s_i = 2), \quad (52)$$

$$\propto p(y_i|s_i = 2, \ldots), \quad (53)$$

$$\propto \theta_2 e^{-\theta_2 y_i}. \quad (54)$$

We then draw s_i from a categorical distribution, $s_i \sim \mathrm{Cat}(\vec{p})$. The categorical distribution is the generalization of the simple Bernoulli trial (or coin flip) to situations where we draw a sample from one of K categories, each category being drawn with probability p_k. That is, $s_i \sim \mathrm{Cat}(s_i; p_1,p_2,\ldots,p_K)$. In this example, our categorical distribution has only two categories,

$$s_i \sim \mathrm{Cat}(s_i; p(s_i = 1|\ldots), p(s_i = 2|\ldots)). \quad (55)$$

Thus, for each data point i we sample the indicator variable according to which component is likely to have generated y_i, conditioned on the current values of θ_1,θ_2.

The last part of our sampling scheme is the mixture weights w_j, for which we will encounter a new conjugate prior model. Note that the indicator variables s_i are drawn from a categorical distribution and that the weights w_j for all the components must sum to 1. Consider the joint posterior distribution of all the mixture weights (here, just two),

$$p(w_1, w_2|\ldots) \propto p(y_N|\ldots)p(w_1, w_2). \quad (56)$$

To sample the mixture weights, we take advantage of the conjugacy between a categorical likelihood and a Dirichlet prior. The Dirichlet distribution is a distribution over a vector of probabilities, which must sum to 1. A K-dimensional Dirichlet distribution is defined on the $(K - 1)$-dimensional simplex, which ensures that the K elements drawn from this distribution will sum to 1. The Dirichlet distribution, with parameters $\alpha_1,\ldots,\alpha_K$ is

$$\mathrm{Dir}(w_1, \ldots, w_K; \alpha_1, \ldots, \alpha_K) = \frac{\Gamma\left(\sum_{j=1}^K \alpha_j\right)}{\prod_{j=1}^K \Gamma(\alpha_j)} \prod_{j=1}^K w_j^{\alpha_j - 1}. \quad (57)$$

This distribution, while perhaps unfamiliar, can be seen as a generalization of the beta-binomial model we used earlier. In that instance, we were interested in a binomial likelihood that quantified the occurrence of binary events. In particular, we wanted to know the parameter θ, the probability of a successful event. In that case, we had two possible outcomes (success or failure), with probability θ and $(1 - \theta)$, respectively. Because the categorical distribution is a generalization of the Bernoulli to situations where we sample from many possible outcomes, the Dirichlet distribution is a generalization of the beta, and quantifies the vector of probabilities of each outcome. Using this as a prior over the weights w_1,w_2 results in a Dirichlet posterior,

$$p(w_1, w_2|\ldots) = \mathrm{Dir}(w_1, w_2; |A_1| + \alpha_1, |A_2| + \alpha_2). \quad (58)$$

With this, we have all the ingredients we need for our Gibbs sampler. For each iteration of the algorithm, we draw random samples for each parameter as

$$\theta_j|\ldots \sim \mathrm{Ga}\left(\theta_j; a + |A_j|, b + \sum_{i \in A_j} y_i\right), \quad (59)$$

$$w_1, w_2|\ldots \sim \mathrm{Dir}(w_1, w_2; |A_1| + \alpha_1, |A_2| + \alpha_2), \quad (60)$$

$$s_i|\ldots \sim \mathrm{Cat}(s_i; p(s_i = 1|\ldots), p(s_i = 2|\ldots)). \quad (61)$$

I next demonstrate this MCMC algorithm with simulated data. The data will be drawn from a mixture of two exponential distributions with timescale parameters $\theta_1 = 1$ and $\theta_2 = 0.01$, and with $w_1 = 0.25$ and $w_2 = 0.75$. Fig. 3 shows a histogram of the logarithm of each data point (25) and a rug plot of all the data is shown below the histogram. When visualized in this way, we can be sure that there are two distinct components within the data. One way to view the task of fitting these data is that we need to decide from which component each data point was drawn and then use the label assignments to estimate each θ_j and w_j. That is, we want the assignments of the s_i to yield high posterior probability. The Gibbs sampling scheme we just laid out will achieve this and the result of this sampler is visualized in Fig. 3. Here, each datapoint is labeled according to which component it is assigned: there is a blue component and a red component. These labels correspond to just one iteration of the Gibbs sampler and thus represent a high posterior explanation of the data, but not the only one. Recall that we want to explore all the values of the parameters that yield good fits to the data, so we want to explore the full posterior distribution. By sampling many label assignments, all of which yield high posterior probability, we marginalize out

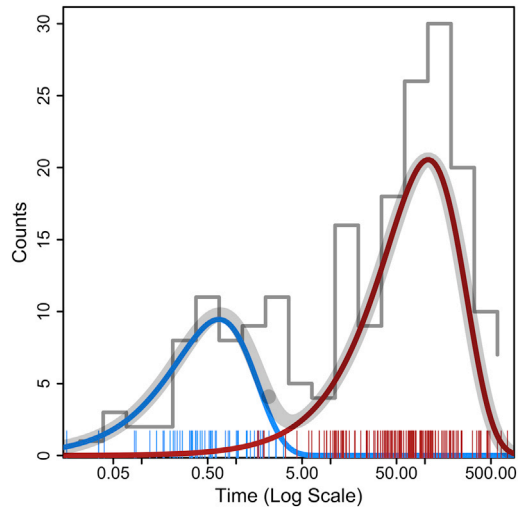

FIGURE 3 An exponential mixture model. Simulated data drawn from a mixture of two exponential distributions. Data are plotted logarithmically for visualization. The result of using a Gibbs sampler to infer the parameters of a two-component mixture model are also shown. Data points are colored corresponding to the component from which they are likely to have been generated. The probability density of each component is shown and the sum of both densities is shown (*shaded*) and matches well with the histogram. To see this figure in color, go online.

the s_i and yield accurate estimates of the total uncertainty in the model parameters in which we are actually interested.

Fig. 4 shows the result of our Gibbs sampler for each of the model parameters of interest: θ_1,θ_2,w_1,w_2. The top row shows the MCMC trajectories for the two dimensions of the Markov chain corresponding to the θ-parameters. Note that on the first iteration of MCMC, the parameters are initialized somewhere arbitrary in the parameter space, but quickly converge to a region of the parameter space that yields high posterior probability. This process of burn-in is discussed in greater detail in the Supporting Material. After the Markov chain has converged, subsequent transitions yield samples from the posterior distribution. For each parameter, the positions of the chain can be aggregated to approximate the marginal posterior distribution of each parameter and this is shown in the second row of Fig. 4. This histogram of MCMC samples approximates the underlying marginal posterior and provides an accurate estimate of the parameter values and their uncertainty. Along with each histogram, the true parameter value is plotted as a vertical line and we see that our posterior distributions, from which we might construct a 95% confidence interval, accurately capture the underlying parameter values. The bottom-half of Fig. 4 similarly shows the MCMC trajectories and marginal posterior distributions for the weight parameters w_1 and w_2. Again, we see that our MCMC estimate of the posterior distribution accurately captures the true parameter values and provides a natural way to quantify parameter confidence.

Using MCMC, we are able to draw samples from a posterior distribution that is unknown to us, and therefore we can effectively estimate posteriors of any dimensionality. For Gibbs sampling, we only need a convenient form for each conditional posterior distribution and the full posterior can be estimated. For many inference settings, this will be adequate because conjugate models have been devised for many kinds of distributions. Even very complex probability models can be deconstructed into simple conditional posteriors for Gibbs sampling. For example, hidden Markov models tend to have many free parameters describing transition dynamics and emission distributions (18). However, with useful data augmentation, the relevant conditional posteriors can be easily calculated and efficient Gibbs sampling schemes devised (19,21). This has already been applied in biophysical settings including modeling ion channel gating (20,24). However, the Gibbs sampler, despite its simplicity and elegance, is inevitably limited to those models where we can calculate conditional posteriors. In some settings, this will not be possible and more general MCMC methods must be used.

Metropolis-Hastings

In many cases, it will not be possible to derive conditional posteriors, because the model parameters may be related to the likelihood only through some complex model function. In these settings, we need to turn to more general forms of MCMC. As a motivating example, we will consider the very general problem of curve-fitting. In common biophysical investigations, some theory is evaluated by its ability to explain carefully controlled experimentation. Our model, with parameters $\theta_1,\theta_2,\ldots,\theta_K$ denoted $\vec{\theta}$, makes a prediction about how some measurable signal might look when examined with respect to some controlled variables. That is, our model prescribes some function $f(\vec{\theta},x)$, which specifies how the observable signal f should depend upon model parameters $\vec{\theta}$ and independent variables x.

As a concrete example, imagine we are modeling the activation of a voltage-gated ion channel. The simplest model would be to assume the channel can exist in a conducting and nonconducting state and the equilibrium between these states is perturbed by transmembrane voltage. Suppose we have measured a conductance-voltage (G-V) curve for this channel and want to fit it to a two-state Boltzmann distribution that quantifies the probability of the channel opening as a function of voltage. In this case, the independent variable is voltage and our two-state model predicts that our G-V curve should follow the form

$$f(a,b,V) = \frac{1}{1+\exp(-(V-a)b)}, \tag{62}$$

where parameters a and b might have some biophysical interpretation. Once we have made some measurements about how the channel activates at various controlled voltages, our goal is to find a good fit between the above

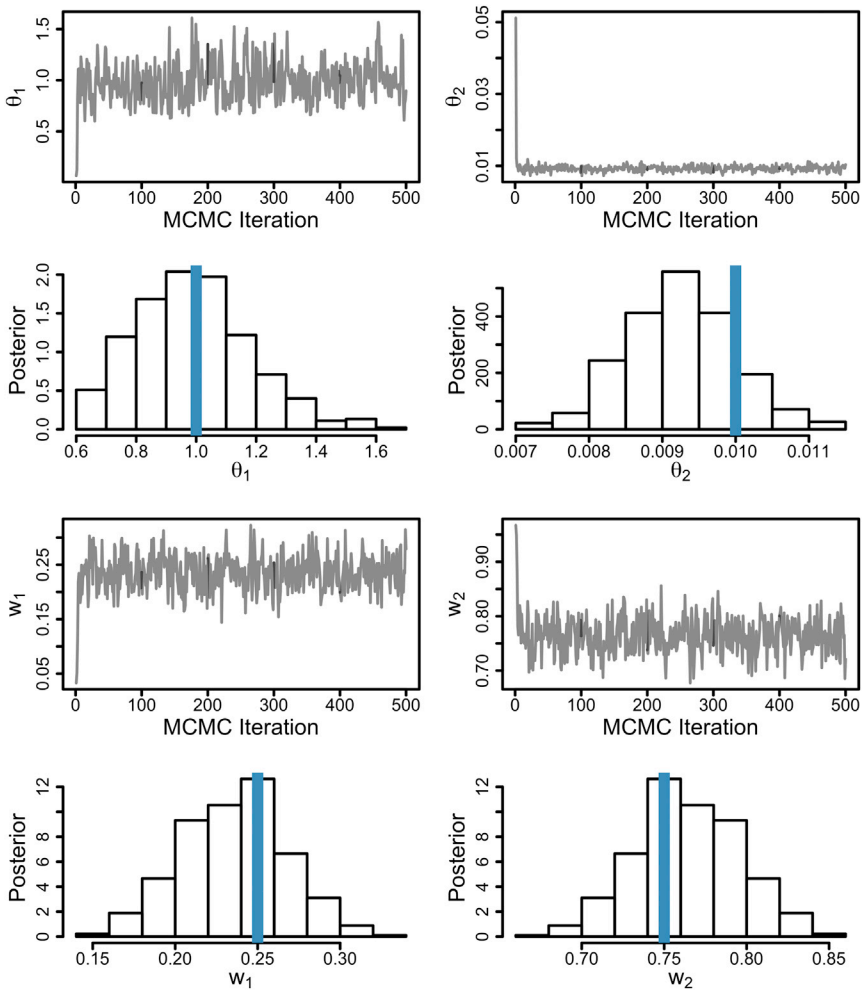

FIGURE 4 Application of Gibbs sampling to the exponential mixture data shown in Fig. 3. For each of the four model parameters (θ_1,θ_2,w_1,w_2), the MCMC trajectories and marginal posterior distributions are shown (probability density). To see this figure in color, go online.

equation and our data. That is, we want to fit the data by exploring the parameter space of a and b until the model prediction adequately matches the measured data. We might embark on this curve-fitting endeavor by searching the parameter space for the point that minimizes the error between the model and the data (13,15) and we would thus accept the resulting values of a and b as indicative of the true values of the underlying biophysical parameters. However, it has been noted that even for simple biophysical models, achieving a good fit to the data provides no guarantee that the recovered parameter estimates are accurate due to the pitfall of parameter nonidentifiability (11,23). Therefore, we might prefer to take a Bayesian approach and seek not just a point estimate of parameters a and b, but instead to quantify the entire posterior distribution, $p(a,b|y)$.

In order to estimate the posterior distribution of our biophysical model, we will rely upon another MCMC method called the Metropolis-Hastings algorithm. First, we decide that our observable signal, which is specified by our model in the form of some $f(\vec{\theta},x)$, is also corrupted by the inevitable presence of experimental noise. For our example, we assume that $f(a,b,V)$ is accompanied by the presence of normally distributed variability. This assumption is not vital, as any noise model could be used, but it seems reasonable in practice and is an assumption at the heart of existing curve-fitting techniques such as error-minimization and maximum-likelihood (22). That is, we assume that each data point y_i arises as a combination of a deterministic function $f(a,b,V)$ and some noisy process with unknown variance,

$$y_i \sim f(a,b,V_i) + N\left(0,\sigma^2\right), \tag{63}$$

where $N(\mu,\sigma)$ denotes a normal distribution with mean μ and standard deviation σ. Given this, our likelihood function is simply a normal distribution centered at f and with variance σ^2,

$$p(y_i|\ldots) = N\left(f(a,b,V_i),\sigma^2\right). \tag{64}$$

We assume that each data point arises from f and some independent and identically distributed noise, so the posterior distribution is

$$p\left(a,b,\sigma^2\middle|y_N\right)\propto\left(\prod_{i=1}^{N}N\left(f(a,b,V_i),\sigma^2\right)\right)p(a)p(b)p\left(\sigma^2\right). \tag{65}$$

When viewed in this way, we can start to guess that we will not be able to use Gibbs sampling here, because our model parameters of interest are related to our likelihood only through a complex function f. Therefore, we have to turn to a more general method of MCMC.

Originally proposed to solve high-dimensional problems in particle physics, what is now known as the Metropolis-Hastings algorithm is a very general tool for estimating probability distributions (16,27). For simplicity, I will describe only a special case of the Metropolis-Hastings method, called the Metropolis random walk. Recall that our posterior distribution of interest has three parameters:

$$\vec{\theta}\;=\;\left\{a,b,\sigma^2\right\}.$$

We will construct a Markov chain whose limiting distribution is the posterior $p(a,b,\sigma^2|y_N)$. Using the Metropolis random walk, this Markov chain evolves with the following rules: At iteration i of the algorithm, the Markov chain is in location θ_i of the parameter space. We generate a proposal movement of the chain by taking a random walk from θ_i to a new location $\tilde{\theta}$. If the proposal point has higher posterior probability than θ_i (i.e., if $p(\tilde{\theta}|y_N)>p(\theta_i|y_N)$), then we accept it and add it to the chain: $\theta_{i+1}=\tilde{\theta}$. If $p(\tilde{\theta}|y_N)<p(\theta_i|y_N)$, then we accept $\tilde{\theta}$ with probability α where α is related to the decrease in posterior probability: $\alpha=p(\tilde{\theta}|y_N)/p(\theta_i|y_N)$. If the proposal is rejected, the Markov chain is extended with its present location, $\theta_{i+1}=\theta_i$. More succinctly, we can describe a single iteration of the Metropolis random walk algorithm as

$$1.\quad\tilde{\theta}\sim\theta_i+N\left(\vec{0},\Sigma\right) \tag{66}$$

$$2a.\quad\text{if}\quad p\left(\tilde{\theta}\middle|y_N\right)>p(\theta_i|y_N):\quad\theta_{i+1}\;=\;\tilde{\theta} \tag{67}$$

$$2b.\quad\text{else draw}\quad u\sim U[0,1] \tag{68}$$

$$\text{if}\quad u<\frac{p\left(\tilde{\theta}\middle|y_N\right)}{p(\theta_i|y_N)}:\quad\theta_{i+1}\;=\;\tilde{\theta} \tag{69}$$

$$\text{else}:\quad\theta_{i+1}\;=\;\theta_i, \tag{70}$$

where Σ is a covariance matrix of our choice that specifies the characteristics of the random-walk portion of the algorithm and $U[0,1]$ is the uniform distribution on the unit interval.

Let us break down, in more detail, what this algorithm does and how it works. The first component is that we attempt to take a random walk in the parameter space and if the proposal point leads to improved posterior probability, then we keep it. This by itself would be a possible (albeit awfully slow) optimization method for finding the maximum of the posterior. But recall that the goal is not to find a point estimate of the parameters, but instead to create a Markov chain that explores the whole parameter space in proportion to posterior probability. Thus, even if $\tilde{\theta}$ leads to a decrease in posterior probability, we still might keep it. And the probability with which we keep it is exactly the ratio of the posterior probabilities of $\tilde{\theta}$ and θ_i. Suppose that θ_i is in an area of high posterior probability and that any random walk away from θ_i is likely to an area of lower posterior probability. We want the chain to be able to visit areas of lower posterior probability and this is exactly what the accept/reject rule achieves. If $p(\tilde{\theta}|y_N)$ is twofold less than $p(\theta_i|y_N)$, then we only accept $\tilde{\theta}$ with probability 1/2. And if $\tilde{\theta}$ is an area of much lower posterior probability, say 100-fold worse, then we would only accept $\tilde{\theta}$ with probability 1/100. In the algorithm above, we draw uniformly distributed random variables and compare them to $p(\tilde{\theta}|y_N)/p(\theta_i|y_N)$ as a particularly simple way of implementing this kind of accept/reject rule. Therefore the chain is able to explore all areas of the parameter space and not just areas of higher posterior probability than its present position. Further, the probability that the chain visits a particular location is exactly the posterior probability at that point in the parameter space and we have successfully constructed a Markov chain whose limiting distribution is the posterior distribution.

It is important to appreciate what this algorithm has gained us. We decided that we would be unable to come up with a simple closed form for the desired posterior, $p(a,b,\sigma^2|y_N)$, or even any conditional distributions for Gibbs sampling. Using the Metropolis random walk, we can estimate the posterior distribution for any model for which we can calculate the likelihood and the prior. This is a major advance. While we may not have a simple form for $p(\theta|y_N)$ for the whole parameter space, if we decide on a particular likelihood and prior, then it is straightforward to compute the posterior probability for any particular parameter value θ_i, $p(\theta_i|y_N)$. In our example, we chose a normal distribution for the likelihood and we can choose any kind of prior that we want for each parameter (Eq. 65). Thus, we easily walk around the parameter space, performing calculations of posterior probability and making accept/reject decisions and the result is samples from the posterior distribution.

Let us return to the example of G-V curves for a demonstration of the Metropolis random walk. At the top of Fig. 5 is a simulated activation curve generated with $a=-50$ mV and $b=0.05$ mV^{-1} and with added Gaussian noise with $\sigma=0.02$. We can use these data to estimate the posterior distribution $p(a,b,\sigma^2|y_N)$ with the algorithm described above. In practice, we can implement the random walk in the full three-dimensional space (as described above), or we can

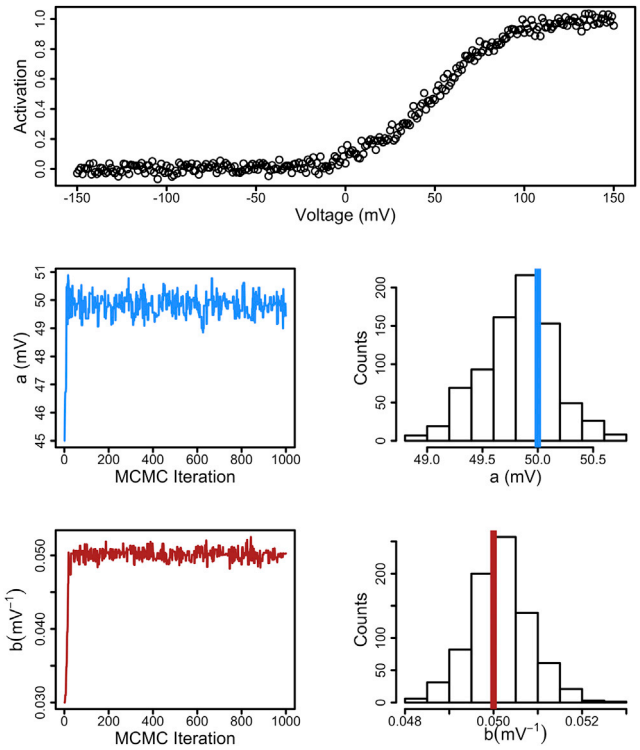

FIGURE 5 Demonstration of Metropolis-Hastings algorithm to analyze ion channel activation data. (*Top panel*) Simulated G-V curve with added Gaussian noise. (*Lower panels*) (*Left*) MCMC trajectories for model parameters a and b; (*right*) marginal posterior distributions of each parameter with the true values shown (*vertical lines*). To see this figure in color, go online.

treat each parameter sequentially (within each iteration) and generate $\tilde{\theta}$ for a single parameter with a one-dimensional random walk. Both approaches will work but there may be slight effects on chain mixing (see the Supporting Material) for some models. For parameters a and b, I have used diffuse normal priors: $p(a) = N(0,100)$ and $p(b) = N(0,5)$. The result of MCMC is shown in Fig. 5 for the two parameters of interest, a and b. At left is the trajectory of each parameter over the course of MCMC and we see that while the parameters are initialized arbitrarily, they quickly converge to areas of higher posterior probability and explore only a small region of the parameter space. At right are histograms of the marginal posterior distributions along with the true parameter values plotted as vertical lines. Using the Metropolis random walk, we are able to recover an accurate estimate of the relevant parameters and their uncertainties. Importantly, to do this we only need to be able to calculate the expectation of the observable signal, $f(V|a,b)$, and the likelihood, $N(f,\sigma^2)$. Therefore, this approach can be used very generally in nearly all modeling endeavors.

Conclusion

The Bayesian methods I have described provide a very general paradigm for parameter inference problems in biophysics. With simple problems, we can calculate posterior distributions directly by using conjugate models. With more complex models, we can turn to computational methods for posterior inference, such as Gibbs sampling or the Metropolis-Hastings algorithm. Additionally, more sophisticated sampling methods exist that will be useful for exploring very high-dimensional posterior distributions (8,17). These methods provide us a foothold to begin exploring more exciting and sophisticated Bayesian models such as Dirichlet process models (10,26) and Gaussian process models (3,7). The use of Bayesian methods for parameter inference gains us three advantages. First, it allows us to express parameter uncertainty as probability, a much more natural notion than that of the frequentist sampling distribution. Second, we gain a simple mechanism to incorporate into the inference process any prior information we might have. Third, and most importantly, Bayesian inference (with the aid of MCMC) gives us a generalizable method of rigorously addressing parameter inference and identifiability for arbitrarily complicated models.

SUPPORTING MATERIAL

Supporting Material, one figure, and one code are available at http://www.biophysj.org/biophysj/supplemental/S0006-3495(15)00303-3.

ACKNOWLEDGMENTS

The author thanks anonymous reviewers for helpful comments on the manuscript.

The author is in the laboratory of Richard W. Aldrich and is supported by National Institutes of Health grant No. R01-NS077821 as well as a predoctoral fellowship from the American Heart Association.

REFERENCES

1. Calderhead, B., M. Epstein, …, M. Girolami. 2013. Bayesian approaches for mechanistic ion channel modeling. *In* In Silico Systems Biology, *Vol. 1021* M. Schneider, editor. Humana Press, Totowa, NJ, pp. 247–272.
2. Colquhoun, D., and A. G. Hawkes. 1981. On the stochastic properties of single ion channels. *Proc. R. Soc. Lond. B Biol. Sci.* 211:205–235.
3. Frigola, R., Y. Chen, and C. Rasmussen. 2014. Variational Gaussian process state-space models. *In* Advances in Neural Information Processing Systems 27 (NIPS 2014). Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors. Curran Associates, Red Hook, NY, pp. 3680–3688.
4. Gelfand, A., and A. Smith. 1990. Sampling-based approaches to calculating marginal densities. *J. Am. Stat. Assoc.* 85:398–409.
5. Geman, S., and D. Geman. 1984. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.* 6:721–741.
6. Geweke, J. 1989. Bayesian inference in econometrics models using Monte Carlo integration. *Econometrica.* 57:1317–1339.
7. Gibson, N., S. Aigrain, …, F. Pont. 2012. A Gaussian process framework for modeling instrumental systematics: application to transmission spectroscopy. *Mon. Not. R. Astron. Soc.* 419:2683–2694.
8. Girolami, M., and B. Calderhead. 2011. Reimann manifold Langevin and Hamiltonian Monte Carlo methods. *J. R. Stat. Soc., B.* 73:123–214.

9. Hines, K. E. 2013. Inferring subunit stoichiometry from single molecule photobleaching. *J. Gen. Physiol.* 141:737–746.
10. Hines, K. E., J. R. Bankston, and R. W. Aldrich. 2015. Analyzing single-molecule time series via nonparametric Bayesian inference. *Biophys. J.* 108:540–556.
11. Hines, K. E., T. R. Middendorf, and R. W. Aldrich. 2014. Determination of parameter identifiability in nonlinear biophysical models: a Bayesian approach. *J. Gen. Physiol.* 143:401–416.
12. Klinke, 2nd, D. J. 2009. An empirical Bayesian approach for model-based inference of cellular signaling networks. *BMC Bioinformatics.* 10:371.
13. Levenberg, K. 1944. A method for the solution of certain problems in least squares. *Q. Appl. Math.* 2:164–168.
14. Loredo, T. 1990. From Laplace to supernova SN1987A: Bayesian inference in astrophysics. *In* Maximum Entropy and Bayesian Methods. P. Fougere, editor. Kluwer Academic Publishers, Dordrecht, The Netherlands, pp. 81–142.
15. Marquardt, D. 1963. An algorithm for least-squares estimation of nonlinear parameters. *J. Soc. Ind. Appl. Math.* 11:431–441.
16. Metropolis, N., A. Rosenbluth, …, E. Teller. 1953. Equation of state calculation by fast computing machines. *J. Chem. Phys.* 21:1087–1092.
17. Neal, R. 2011. MCMC using Hamiltonian dynamics. *In* Handbook and Markov Chain Monte Carlo. S. Brooks, A. Gelman, G. Jones, and X. Meng, editors. Chapman and Hall/CRC, Boca Raton, FL.
18. Rabiner, L. 1989. A tutorial on hidden Markov models and select applications in speech recognition. *Proc. IEEE.* 77:257–286.
19. Robert, C., G. Celeux, and J. Diebolt. 1993. Bayesian estimation of hidden Markov chains: a stochastic implementation. *Stat. Probab. Lett.* 16:77–83.
20. Rosales, R. A. 2004. MCMC for hidden Markov models incorporating aggregation of states and filtering. *Bull. Math. Biol.* 66:1173–1199.
21. Scott, S. 2002. Bayesian methods for hidden Markov models. *J. Am. Stat. Assoc.* 97:337–351.
22. Seber, G., and C. Wild. 2003. Nonlinear Regression. Wiley Interscience, Hoboken, NJ.
23. Siekmann, I., J. Sneyd, and E. J. Crampin. 2012. MCMC can detect nonidentifiable models. *Biophys. J.* 103:2275–2286.
24. Siekmann, I., L. E. Wagner, 2nd, …, J. Sneyd. 2011. MCMC estimation of Markov models for ion channels. *Biophys. J.* 100:1919–1929.
25. Sigworth, F. J., and S. M. Sine. 1987. Data transformations for improved display and fitting of single-channel dwell time histograms. *Biophys. J.* 52:1047–1054.
26. Teh, Y., M. Jordan, …, D. Blei. 2006. Hierarchical Dirichlet processes. *J. Am. Stat. Assoc.* 101:1566–1581.
27. Tierney, L. 1994. Markov chains for exploring posterior distributions. *Ann. Stat.* 22:1701–1728.
28. Ulbrich, M. H., and E. Y. Isacoff. 2007. Subunit counting in membrane-bound proteins. *Nat. Methods.* 4:319–321.