

# Model-based decoding, information estimation, and change-point detection techniques for multi-neuron spike trains

Jonathan W. Pillow<sup>1</sup>, Yashar Ahmadian<sup>2</sup> and Liam Paninski<sup>2</sup>

<sup>1</sup> Center for Perceptual Systems, The University of Texas at Austin

<sup>2</sup> Department of Statistics and Center for Theoretical Neuroscience,  
Columbia University

pillow@mail.utexas.edu, yashar@stat.columbia.edu, liam@stat.columbia.edu

July 2, 2010

One of the central problems in systems neuroscience is to understand how neural spike trains convey sensory information. Decoding methods, which provide an explicit means for reading out the information contained in neural spike responses, offer a powerful set of tools for studying the neural coding problem. Here we develop several decoding methods based on point-process neural encoding models, or “forward” models that predict spike responses to stimuli. These models have concave log-likelihood functions, which allow for efficient maximum-likelihood model fitting and stimulus decoding. We present several applications of the encoding-model framework to the problem of decoding stimulus information from population spike responses: (1) a tractable algorithm for computing the *maximum a posteriori* (MAP) estimate of the stimulus, the most probable stimulus to have generated an observed single- or multiple-neuron spike train response, given some prior distribution over the stimulus; (2) a Gaussian approximation to the posterior stimulus distribution that can

be used to quantify the fidelity with which various stimulus features are encoded; (3) an efficient method for estimating the mutual information between the stimulus and the spike trains emitted by a neural population; and (4) a framework for the detection of change-point times (e.g. the time at which the stimulus undergoes a change in mean or variance), by marginalizing over the posterior stimulus distribution. We provide several examples illustrating the performance of these estimators with simulated and real neural data.

## 1 Introduction

The neural decoding problem is a fundamental question in computational neuroscience (Rieke et al., 1997): given the observed spike trains of a population of cells whose responses are related to the state of some behaviorally-relevant signal  $\mathbf{x}$ , how can we estimate, or “decode,”  $\mathbf{x}$ ? Solving this problem experimentally is of basic importance both for our understanding of neural coding and for the design of neural prosthetic devices (Donoghue, 2002). Accordingly, a rather large literature now exists on developing and applying decoding methods to spike train data, both in single-cell and population recordings.

This literature can be roughly broken down into two parts, in which the decoding algorithm is based on either 1) regression techniques, or 2) Bayesian methods. Following the influential work of (Bialek et al., 1991), who proposed an optimal linear decoder posed as a version of the Wiener-Hopf problem, the last two decades have seen a great number of papers employing regression methods, typically multiple linear regression in the time or frequency domain (Theunissen et al., 1996; Haag and Borst, 1997; Warland et al., 1997; Salinas and Abbott, 2001; Serruya et al., 2002; Nicolelis et al., 2003; Mesgarani et al., 2009); see also the earlier work of (Humphrey et al., 1970). Elaborations on this idea

include using nonlinear terms in the regression models, e.g. polynomial terms, as in the Volterra model (Marmarelis and Marmarelis, 1978; Bialek et al., 1991), or using neural network (Warland et al., 1997) or kernel regression (Shpigelman et al., 2003; Eichhorn et al., 2004) techniques. These methods tend to be quite computationally efficient, but are not guaranteed to perform optimally for plausible models of the encoding process, and do not explicitly incorporate prior information about the stimulus domain.

On the other hand are decoding algorithms based on Bayes' rule, in which the *a priori* distribution of the signal to be decoded is combined with a "forward" or "encoding" model describing the probability of the observed spike train, given the signal (Fig. 1). The resulting Bayes estimate is by construction optimal, assuming that the prior distribution and encoding model are correct. This estimate also comes with natural "error bars", measures of how confident we should be about our predictions, arising from the posterior distribution over the stimulus given the response. Decoding therefore serves as a means for probing which aspects of the stimulus are preserved by the response, and as a tool for comparing different encoding models. For example, we can decode a spike train using different models (e.g., including vs. ignoring spike-history effects) and examine which encoding model allows us to best decode the true stimulus (Pillow et al., 2005; Pillow et al., 2008). Such a test may in principle give a different outcome than a comparison which focuses on two encoding models' accuracy in predicting spike train responses (Latham and Nirenberg, 2005).

However, computing this Bayes-optimal solution can present computational difficulties. For example, computing the optimal Bayesian estimator under the squared-error cost function requires the computation of a conditional expectation of the signal  $\mathbf{x}$  given the observed spike response,  $\mathbb{E}[\mathbf{x}|\text{spikes}]$ , which in turn requires that we compute  $d$ -dimensional integrals (where  $d = \dim(\mathbf{x})$ ). Thus, most previous work on Bayesian decoding of spike trains has either focused on low-dimensional signals  $\mathbf{x}$  (Sanger, 1994; Maynard et al.,

1999; Abbott and Dayan, 1999; Karmeier et al., 2005) or on situations in which recursive techniques may be used to perform these conditional expectation computations efficiently, either using approximate techniques related to the classical Kalman filter (Zhang et al., 1998; Brown et al., 1998; Barbieri et al., 2004; Wu et al., 2004; Srinivasan et al., 2006; Wu et al., 2009; Yu et al., 2007; Paninski et al., 2010) or variants of the particle filtering algorithm (Brockwell et al., 2004; Kelly and Lee, 2004; Shoham et al., 2005; Ergun et al., 2007; Brockwell et al., 2007), which is exact in the limit of an infinite number of particles. While this recursive approach is quite powerful, unfortunately its applicability is limited to cases in which the joint distribution of the signal  $\mathbf{x}$  and the spike responses has a certain Markov tree decomposition (e.g., a hidden Markov model or state-space representation) (Jordan, 1999).

Here we explore conditions under which certain well-known approximations to the posterior density are applicable without any such tree-decomposition assumptions, and which remain tractable even when the stimulus  $\mathbf{x}$  is very high-dimensional. The idea is to compute the *maximum a posteriori* (MAP) estimate  $\mathbf{x}_{\text{map}}$ . This estimate is Bayesian in the sense that it incorporates knowledge of both the prior distribution  $p(\mathbf{x})$  and the likelihood  $p(\mathbf{r}|\mathbf{x})$ , which is the conditional probability of the observed spike train responses  $\mathbf{r}$  given the stimulus  $\mathbf{x}$ .<sup>1</sup> However, computing  $\mathbf{x}_{\text{map}}$  requires only that we perform a maximization of the posterior, instead of an integration. In the cases we examine here, the posterior is often easier to exactly maximize than to integrate; we will discuss related efficient methods for integration of the posterior in a companion paper (Ahmadian et al., 2010).

We begin by introducing the forward encoding model we use to calculate the encoding distribution  $p(\mathbf{r}|\mathbf{x})$ ; this model incorporates stimulus dependence and spike-history effects

---

<sup>1</sup>The MAP estimate is also Bayes-optimal under a “zero-one” loss function, which rewards only the correct estimate of the stimulus and penalizes all incorrect estimates with a fixed penalty.

(such as refractoriness and adaptation), and may also include multi-neuronal terms corresponding to excitatory or inhibitory effects the activity of one cell has upon another. The model has a key concavity property that makes maximization in  $\mathbf{x}$  highly tractable, and moreover leads to an accurate and simple Gaussian approximation to the posterior  $p(\mathbf{x}|\mathbf{r})$ , which allows us to quantify the fidelity with which various stimulus features are encoded. Finally, this approximation can be integrated analytically, which allows us to: 1) efficiently estimate the mutual information  $I[\mathbf{x}|\mathbf{r}]$  between the high-dimensional stimulus  $\mathbf{x}$  and the response  $\mathbf{r}$ , and 2) optimally detect change-points: times at which some property of the distribution from which the stimuli are drawn (e.g., the mean or variance) undergoes a change. Each of these applications are illustrated with several examples below.

## 2 Encoding model

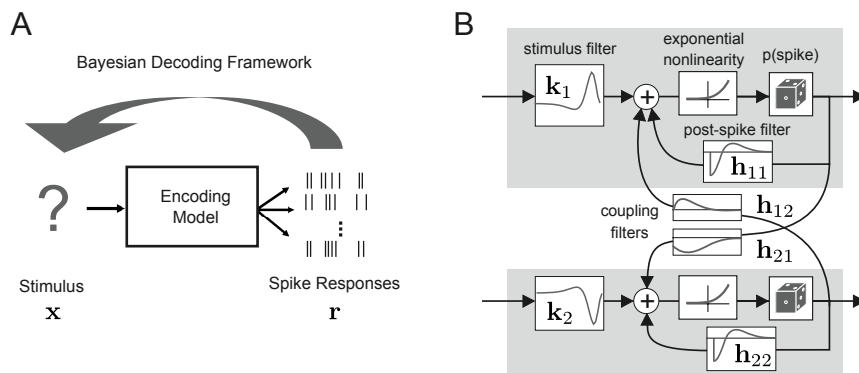


Figure 1: Illustration of Bayesian decoding paradigm and point process encoding model. **(A)** Bayesian decoding performs inference about the stimulus  $\mathbf{x}$  using the observed spike times  $\mathbf{r}$  and a specified encoding model. **(B)** Schematic of the Generalized Linear Model (GLM) encoding model used for the decoding examples shown in this paper. The model parameters  $\{\mathbf{k}_i\}$  and  $\{\mathbf{h}_{ij}\}$  are parameters relating the stimulus and spike-history to the instantaneous spike probability, and can be fit using maximum likelihood. For decoding applications, the fitted model provides the stimulus likelihood,  $p(\mathbf{r}|\mathbf{x})$ , which is combined with the prior  $p(\mathbf{x})$  to form the posterior  $p(\mathbf{x}|\mathbf{r})$ , which is maximized to obtain the estimate  $\hat{\mathbf{x}}_{MAP}$ .

We model a neural spike train as a point process generated by a generalized linear model (GLM) (Brillinger, 1988; McCullagh and Nelder, 1989; Paninski, 2004; Truccolo et al., 2005). This model class has been discussed extensively elsewhere. Briefly, this class is a natural extension of the linear-nonlinear-Poisson (LNP) model used in reverse correlation analyses (Simoncelli et al., 2004), with close connections to biophysical models such as the integrate-and-fire model (Paninski et al., 2004b; Paninski, 2006). It has been applied in a variety of experimental settings (Brillinger, 1992; Dayan and Abbott, 2001; Chichilnisky, 2001; Paninski et al., 2004a; Truccolo et al., 2005; Pillow et al., 2008; Gerwinn et al., 2008; Stevenson et al., 2009; Truccolo et al., 2010). Figure 1B shows a schematic of this model’s parameters, for two coupled neurons. The model is summarized as:

$$\lambda_i(t) = f\left(\mathbf{k}_i \cdot \mathbf{x}(t) + \sum_{j,\alpha} \mathbf{h}_{ij}(t - t_{j\alpha}) + b_i\right), \quad (1)$$

where  $\lambda_i(t)$  denotes the conditional intensity (or instantaneous firing rate) of the  $i$ ’th cell at time  $t$ ,  $\mathbf{k}_i$  is the cell’s linear receptive field,  $b_i$  is an additive constant determining the baseline spike rate,  $\mathbf{h}_{ij}(t)$  a post-spike effect from the  $j$ ’th to the  $i$ ’th observed neuron in the population of cells, and  $t_{j\alpha}$  is the  $\alpha$ ’th spike from the  $j$ ’th neuron, where the sum is taken over all past spike times  $t_{j\alpha} < t$ . The  $\mathbf{h}_{ii}(t)$  term (corresponding to the  $i$ ’th cell’s own spikes) can be thought of as a linear filter acting on the cell’s own spike history, and can account for refractory effects, burstiness, firing-rate adaptation, etc., depending on the shape of  $\mathbf{h}_{ii}(t)$ . The  $\mathbf{h}_{ij}(t)$  terms from the other cells correspond to interneuronal interaction effects, and may be excitatory and/or inhibitory.

Under one physiological interpretation of this model, known as a “spike-response” or “soft-threshold integrate-and-fire” model, the net linearly filtered input (i.e., the argument to  $f$ ) is regarded as the leaky integral of input currents, which specifies a non-dimensionalized intracellular voltage;  $f$  converts this voltage to an instantaneous probabil-

ity of spiking, which increases monotonically as a function of the height of voltage above threshold (Plesser and Gerstner, 2000; Paninski, 2006).

If the function  $f(u)$  is convex and  $\log f(u)$  is concave in the argument  $u$  (for example,  $f(u) = \exp(u)$ ), then the log-likelihood function

$$L(\mathbf{x}, \theta) = \log p(\mathbf{r}|\mathbf{x}, \theta) = \sum_{i,\alpha} \log \lambda_i(t_{i\alpha}) - \sum_i \int_0^T \lambda_i(t) dt + \text{const.} \quad (2)$$

is guaranteed to be a concave function of either the stimulus  $\mathbf{x}$  or the model parameters  $\theta = \{\{\mathbf{k}_i\}, \{\mathbf{h}_{ij}\}, \{b_i\}\}$ , no matter what spike data  $\mathbf{r}$  are observed (Paninski, 2004), where  $[0, T]$  is the time interval on which the responses are observed.<sup>2</sup> Log-concavity with respect to model parameters makes it easy to fit the model, since concave functions on convex parameter spaces have no non-global local maxima. We can therefore use simple gradient ascent techniques (e.g. conjugate gradients or Newton’s method (Press et al., 1992)) to compute the maximum likelihood estimate of the model parameters  $\hat{\theta}$ .<sup>3</sup>

One should note that this restriction on the nonlinearity  $f(\cdot)$  (i.e., that  $f(\cdot)$  must be convex and log-concave) is nontrivial. In particular, two important cases are ruled out:

1. saturating nonlinearities (e.g.,  $f(x) = \tanh(x)$ );
2. non-monotonic nonlinearities (e.g., the squaring nonlinearity  $f(x) = x^2$ ).

The former restriction turns out to be relatively minor, since we may enforce saturating

---

<sup>2</sup>Note that the log-likelihood function is separately, not jointly, concave in  $\mathbf{x}$  and the model parameters; that is,  $L$  is concave in the stimulus  $\mathbf{x}$  for any fixed data  $\mathbf{r}$  and parameters  $\theta$ , and concave in the parameters  $\theta$  for any fixed observed  $\mathbf{r}$  and  $\mathbf{x}$ .

<sup>3</sup>Note that the GLM is not the only model with this useful concavity property; for example, the more biophysically-motivated noisy leaky integrate-and-fire (IF) model has the same property (Pillow et al., 2004; Paninski et al., 2004b), and all of the methods introduced here apply equally well in the IF context. However, for simplicity, we will restrict our attention to the GL model here.

firing rates by choosing the spike-history function  $\mathbf{h}_{ii}(\cdot)$  to be strongly inhibitory for small times (enforcing an absolute refractory period) (Berry and Meister, 1998; Paninski, 2004). The latter restriction, on the other hand, is more severe; see the end of section 7.1 for further discussion of this point.

### 3 MAP estimation

To compute  $\mathbf{x}_{\text{map}}$  we need to maximize the posterior over the stimulus given the data:

$$p(\mathbf{x}|\mathbf{r}) = \frac{1}{Z}p(\mathbf{r}|\mathbf{x})p(\mathbf{x})$$

as a function of  $\mathbf{x}$ , where  $Z$  is a normalizing constant that does not depend on  $\mathbf{x}$ . This is equivalent to maximizing

$$\log p(\mathbf{x}|\mathbf{r}) = \log p(\mathbf{r}|\mathbf{x}) + \log p(\mathbf{x}) + \text{const.} \quad (3)$$

As discussed above, the log-likelihood term  $\log p(\mathbf{r}|\mathbf{x})$  is concave in  $\mathbf{x}$ , for any observed spike data  $\mathbf{r}$ . Since the sum of two concave functions is itself concave, it is clear that this optimization problem will be tractable whenever the log-prior term  $\log p(\mathbf{x})$  is also a concave function of  $\mathbf{x}$  (Paninski, 2004): in this case, any ascent algorithm is guaranteed to return the optimal solution

$$\mathbf{x}_{\text{map}} \equiv \arg \max_{\mathbf{x}} \log p(\mathbf{x}|\mathbf{r}),$$

since  $\mathbf{x}$  lies within a convex set (the  $d$ -dimensional vector space). Note that this optimal solution  $\mathbf{x}_{\text{map}}$  is in general a nonlinear function of the data  $\mathbf{r}$ .



We should emphasize that log-concavity of the stimulus distribution  $p(\mathbf{x})$  is a restrictive condition (Paninski, 2005). For example, log-concave distributions must have tails that decrease at least exponentially quickly, ruling out “heavy-tailed” prior distributions with infinite moments. However, the class of log-concave distributions is quite large, including (by definition) any distribution of the form

$$p(\mathbf{x}) = \exp(Q(\mathbf{x}))$$

for some concave function  $Q(\mathbf{x})$ ; for example, the exponential, triangular, uniform, and multivariate Gaussian (with arbitrary mean and covariance) distributions may all be written in this form. In particular, any experiment based on the “white noise” paradigm, in which a Gaussian signal of some mean and a white power spectrum (or more generally, *any* power spectrum) is used to generate stimuli (see e.g. (Marmarelis and Marmarelis, 1978) or (Rieke et al., 1997) for many examples), may be easily analyzed in this framework. Of course, in principle we may still compute  $\mathbf{x}_{\text{map}}$  in the case of non-concave log-priors; the point is that ascent techniques might not return  $\mathbf{x}_{\text{map}}$  in this case, and that therefore computing the true global optimizer  $\mathbf{x}_{\text{map}}$  may not be tractable in this more general setting.

### 3.1 Numerical implementation

We have shown that ascent-based methods will succeed in finding the true  $\mathbf{x}_{\text{map}}$ , but it remains to show that these operations are tractable and can be performed efficiently. To compute  $\mathbf{x}_{\text{map}}$ , we may employ Newton-Raphson or conjugate-gradient ascent methods with analytically computed gradients and Hessian, which can be specified as follows.

Let  $K_i$  denote the matrix implementing the linear transformation  $(K_i \mathbf{x})_t = \mathbf{k}_i \cdot \mathbf{x}(t)$ , the projection of the stimulus onto the  $i$ 'th neuron's stimulus filter. (If  $\mathbf{k}_i$  is a purely temporal filter, then  $K_i$  is a Toeplitz matrix with a shifted copy of  $\mathbf{k}_i$  in each row).  $K_i \mathbf{x}$  is thus the vector of stimulus-dependent inputs into the nonlinearity for neuron  $i$  over the time window in question. Combining our expressions for the log-likelihood (eq. 2) and the log-posterior (eq. 3), it is clear we need to minimize an expression of the form

$$-\log p(\mathbf{x}|\mathbf{r}) = -\sum_{i,t} r_i(t) \log f_t((K_i \mathbf{x})_t) + \sum_{i,t} f_t((K_i \mathbf{x})_t) dt - \log p(\mathbf{x}), \quad (4)$$

where for convenience we have abbreviated the time-varying stimulus-dependent rate  $f_t[(K_i \mathbf{x})_t] = f[(K_i \mathbf{x})_t + \sum_{j,\alpha} h_{ij}(t - t_{j\alpha}) + b_i]$ . If we let  $\mathbf{r}_i$  denote a (discretized) vector representation of the  $i$ 'th neuron's spike train, and let  $\mathbf{f}_i$  and  $\mathbf{g}_i$  denote vector versions of  $f_t((K_i \mathbf{x})_t)$  and  $\log f_t((K_i \mathbf{x})_t)$ , respectively, we can rewrite this as

$$-\log p(\mathbf{x}|\mathbf{r}) = \sum_i (-\mathbf{r}_i^T \mathbf{g}_i + \mathbf{1}^T \mathbf{f}_i dt) - \log p(\mathbf{x}). \quad (5)$$

The contribution to the gradient and Hessian (second-derivative matrix) of the negative log-likelihood from a single neuron is therefore given by simple matrix multiplications:

$$\nabla_i = K_i(\mathbf{f}'_i dt - \mathbf{r}_i \cdot \mathbf{g}'_i) \quad (6)$$

$$J_i = K_i^T \text{diag}[\mathbf{f}''_i dt - \mathbf{r}_i \cdot \mathbf{g}''_i] K_i, \quad (7)$$

where  $\mathbf{f}'_i$ ,  $\mathbf{g}'_i$  and  $\mathbf{f}''_i$ ,  $\mathbf{g}''_i$  denote the first and second derivatives of  $\mathbf{f}_i$  and  $\mathbf{g}_i$  with respect to their arguments, and  $\mathbf{a} \cdot \mathbf{b}$  denotes pointwise multiplication of the vectors  $\mathbf{a}$  and  $\mathbf{b}$ . Note that, by the simultaneous convexity and log-concavity of  $f(\cdot)$ , and the nonnegativity of  $r_i(t)$ , the vector  $\mathbf{f}''_i dt - \mathbf{r}_i \cdot \mathbf{g}''_i$  is non-negative, ensuring that the Hessian  $J_i$  is positive semi-definite and therefore that the negative log-likelihood is convex (i.e., the log-likelihood is

concave). To obtain the gradient  $\nabla$  and Hessian  $J$  of the full posterior, these terms are simply summed over neurons and added to the gradient and Hessian of the negative log-prior. For a Gaussian prior with mean  $\mu$  and covariance  $\Lambda$ , the gradient and Hessian of the log-prior are given by  $\Lambda^{-1}(\mathbf{x} - \mu)$  and  $\Lambda^{-1}$ , respectively.

While in general it takes  $O(d^2)$  steps to compute the Hessian of a  $d$ -dimensional function (where  $d = \dim(\mathbf{x})$  here), the Hessian of  $\log p(\mathbf{x}|\mathbf{r})$  may be computed much more quickly. Most importantly, the log-likelihood Hessian  $J_i$  (eq. 7) is a banded matrix, with the width of the band equal to the length of the filter  $\mathbf{k}_i$ . Additionally, the Hessian of the log-prior can be computed easily in many important cases. For example, in the case of Gaussian stimuli, this Hessian is constant as a function of  $\mathbf{x}$  and can be precomputed just once. Thus in fact the amortized computational cost of this Hessian term is just  $O(d)$ , instead of the  $O(d^2)$  time required more generally.

Optimization via Newton-Raphson or conjugate gradient ascent requires  $O(d^3)$  time in general (Press et al., 1992). In special cases, however, we may reduce this significantly. For example, Newton’s optimization method requires that we solve equations of the form  $J\mathbf{x} = \nabla$  for an unknown vector  $\mathbf{x}$ , where  $J$  is the Hessian matrix and  $\nabla$  is the gradient of the negative log-posterior. If the Hessian of the log-prior in our case is banded, then the full Hessian  $J$  will be banded (since the likelihood Hessian  $J$  is banded, as discussed above), and therefore each Newton iteration can be performed in  $O(T)$  time, where  $T$  is the temporal duration of the stimulus  $\mathbf{x}$  (and therefore  $d$  is in most cases proportional to  $T$ ); see (Fahrmeir, 1992; Paninski et al., 2010) for further discussion. We have found empirically that the number of Newton iterations does not scale appreciably with  $T$ , so in these cases the full optimization requires just  $O(T)$  time. These methods may be adapted to handle some settings where a convex constrained optimization over  $\mathbf{x}$  is required (Koyama and Paninski, 2009), or where the log-prior Hessian is not banded, but its inverse is (Ahmadian

et al., 2010). We discuss several related examples in more detail below; see especially Figs. 5-7.

### 3.2 Perturbative analysis and Gaussian approximation

The estimate  $\mathbf{x}_{\text{map}}$  proves to be a good decoder of spike train data in a variety of settings, and the ability to tractably perform optimal nonlinear signal reconstruction given the activity of ensembles of interacting neurons is quite useful per se. However, computing  $\mathbf{x}_{\text{map}}$  gives us easy access to several other important and useful quantities. In particular, we would like to quantify the uncertainty in our estimates. One easy way to do this is by perturbing  $\mathbf{x}_{\text{map}}$  slightly in some direction  $\mathbf{y}$  (say,  $\mathbf{x}_{\text{map}} + \epsilon\mathbf{y}$ , for some small positive scalar  $\epsilon$ ) and computing the ratio of posteriors at these two points  $p(\mathbf{x}_{\text{map}}|\mathbf{r})/p(\mathbf{x}_{\text{map}} + \epsilon\mathbf{y}|\mathbf{r})$ , or equivalently, the difference in the log-posteriors  $\log p(\mathbf{x}_{\text{map}}|\mathbf{r}) - \log p(\mathbf{x}_{\text{map}} + \epsilon\mathbf{y}|\mathbf{r})$ . If the posterior changes significantly with the perturbation  $\epsilon\mathbf{y}$ , then this perturbation is highly “detectable”; it is highly discriminable from  $\mathbf{x}_{\text{map}}$ . Conversely, if the change in the posterior is small, then it is difficult to discriminate between  $\mathbf{x}_{\text{map}}$  and  $\mathbf{x}_{\text{map}} + \epsilon\mathbf{y}$  on the basis of the data  $\mathbf{r}$ . We can expect our estimate  $\mathbf{x}_{\text{map}}$  to be highly variable in this direction, and the corresponding confidence interval in this direction to be wide.

Assuming the log-posterior  $\log p(\mathbf{x}_{\text{map}}|\mathbf{r})$  is smooth, for sufficiently small stimulus perturbations  $\epsilon$  a second-order expansion suffices to approximate the log-posterior:

$$\log p(\mathbf{x}_{\text{map}} + \epsilon\mathbf{y}|\mathbf{r}) = \log p(\mathbf{x}_{\text{map}}|\mathbf{r}) - \frac{\epsilon^2}{2}\mathbf{y}^T J\mathbf{y} + o(\epsilon^2) \quad (8)$$

where  $J$  denotes the Hessian of  $-\log p(\mathbf{x}|\mathbf{r})$  with respect to  $\mathbf{x}$ , evaluated at  $\mathbf{x}_{\text{map}}$ . The expansion lacks a first-order term since the first derivative is (by definition) zero at the

optimizer  $\mathbf{x}_{\text{map}}$ . The quadratic term may be most easily interpreted by computing the eigenvectors of  $J$  (Huys et al., 2006): eigenvectors corresponding to large eigenvalues represent stimulus directions  $\mathbf{y}$  along which the curvature of the posterior is large, i.e. directions along which perturbations are highly discriminable, and those corresponding to small eigenvalues represent directions that are only weakly discriminable.

This second-order description of the log-posterior corresponds to a Gaussian approximation of the posterior (known in the statistics literature as a ‘‘Laplace approximation’’ (Kass and Raftery, 1995)):

$$p(\mathbf{x}|\mathbf{r}) \approx \mathcal{N}(\mathbf{x}_{\text{map}}, C), \quad (9)$$

where the mean of this Gaussian is the MAP estimate  $\mathbf{x}_{\text{map}}$  and the covariance matrix is  $C = J^{-1}$ ; we may use this approximate posterior covariance matrix  $C$  to quantify our uncertainty about  $\mathbf{x}$  remaining after the spiking data  $\mathbf{r}$  have been observed. In general applications, of course, such a Gaussian approximation is not justified. However, in our case, we know that  $p(\mathbf{x}|\mathbf{r})$  is always unimodal (since any concave, or log-concave, function is unimodal (Boyd and Vandenberghe, 2004)) and at least continuous on its support (again by log-concavity). If the nonlinearity  $f(u)$  and the log-prior  $\log p(\mathbf{x})$  are smooth functions of their arguments  $u$  and  $\mathbf{x}$  respectively, then the log-posterior  $\log p(\mathbf{x}|\mathbf{r})$  is necessarily smooth as well. In this case, the Gaussian approximation is often well-justified, although of course the posterior will never be exactly Gaussian; see, e.g., (Minka, 2001; Yu et al., 2007; Koyama and Paninski, 2009) for a discussion of alternate Gaussian approximations based on expectation propagation. See Fig. 3 for some comparisons of the true posterior and this Gaussian approximation.

### 3.3 MAP decoding: examples

Figures 2–7 show several applications of MAP decoding. Figure 2 provides a straightforward example of MAP estimation of a 30-sample stimulus using either a 2-cell (A) or a 20-cell (B) simulated population response. In this case, the stimulus (black trace) consisted of 1 second (30 samples) of Gaussian white noise, refreshed every 33ms. Spike responses (dots) were generated from a point process encoding model (Fig. 1), with parameters fit to responses of macaque retinal ganglion ON and OFF cells (Pillow et al., 2008). Gray and black dotted traces show the MAP estimate computed using the 2-cell and 20-cell population response, respectively. The shaded regions (2C, D) show one standard deviation of the marginal posterior uncertainty about each stimulus value, computed as the square root of the diagonal of the inverse Hessian  $J^{-1}$  (i.e., the square root of the Hessian-based approximation to the marginal variance).

Note, however, that this shading provides an incomplete picture of our uncertainty about the stimulus. The posterior  $p(\mathbf{x}|\mathbf{r})$  is a probability distribution in 30 dimensions, and its curvature along each of these dimensions is captured by the Hessian  $J$ . Eigenvectors associated with the smallest and largest eigenvalues of  $J^{-1}$  correspond to directions in this space along which the distribution is most and least tightly constrained; these can be conceived as “features” that are encoded with the “best” and “worst” fidelity by the population response, respectively. (Obviously, these features are much more interpretable if they correspond to “isolated” eigenvalues that are well separated at the bottom or top of the distribution; otherwise, linear combinations of features associated with nearby eigenvalues will be encoded with approximately the same fidelity). For both populations, the eigenvalues saturate at 1, which is also the stimulus prior variance, reflecting the fact that the response carries no information about the stimulus along these axes (which arises

from a high-temporal-frequency cutoff in retinal ganglion cell responses).

Implicit in this analysis of coding fidelity is the Gaussian approximation to the posterior introduced above (eq. 9). If the shape of the true posterior is poorly approximated by a Gaussian (i.e., the log-posterior is poorly approximated by a quadratic), then the Hessian does not fully describe our posterior uncertainty. Conversely, if the posterior *is* approximately Gaussian, then the posterior maximum is also the posterior mean, meaning that  $\mathbf{x}_{\text{map}}$  closely matches the Bayes' Least Squares (BLS) estimator, which is optimal under mean-squared error loss. Figure 3 shows a comparison between the true posterior and the Gaussian approximation around  $\mathbf{x}_{\text{map}}$  for the 2-cell and 20-cell population responses shown in figure 2. Although log-scaling of the vertical axis (bottom row) reveals discrepancies in the tails of the distribution, the Gaussian approximation generally provides a close match to the shape of the central peak and an accurate description of the bulk of the probability mass under the posterior (top row). Analysis with more computationally-expensive Monte Carlo methods indicate that  $\mathbf{x}_{\text{map}}$  is usually quite closely matched to the posterior mean if the prior  $p(\mathbf{x})$  is smooth (see (Ahmadian et al., 2010) for details).

Next, we examined the role of the stimulus prior in MAP decoding, using stimuli generated from a non-independent Gaussian prior. Figure 4 shows an example in which a Gaussian stimulus was drawn to have a power spectrum that falls as  $1/F$  ("pink noise"), meaning that low frequencies predominate and the stimulus is strongly correlated at short timescales. The true stimulus is plotted in black, and the left panel shows  $\mathbf{x}_{\text{map}}$  computed from the response of two neurons, either assuming that the stimulus prior was independent, but with the correct stationary marginal variance (top), or using the correct  $1/F$  prior (bottom). Note that the likelihood term is identical in both cases, i.e., only the prior gives rise to the difference between the two estimates. The right panel shows a comparison using the same stimulus decoded from the responses of 10 ON and 10 OFF cells. This illustrates

that although both estimates converge to the correct stimulus as the number of neurons increases, the prior still gives rise to a significant difference in decoding performance.

In our next example, presented in Figs. 5 and 6, we consider MAP decoding of a binary 64-pixel, 120 Hz white noise movie presented to a group of 27 retinal ganglion cells, based on the experimentally recorded spike trains of these cells. (See (Pillow et al., 2008) for experimental details and a full description of the estimated encoding model parameters here.) On each movie frame, each pixel was independently and randomly given high or low contrast ( $\pm c$ ) with equal probability. The true distribution describing this stimulus ensemble is therefore binary, and is not log-concave. To exploit the efficient optimization methods discussed here, we used the closest log-concave relaxation of this binary prior (specifically, a uniform prior on the interval  $[-c, c]$ ) as a surrogate prior distribution for decoding. Finding the MAP with this prior corresponds to a (concave) constrained optimization problem. To handle this constrained problem using the  $O(T)$  decoding methods explained in Sec. 3.1, we used an interior-point (aka “barrier”) method (Boyd and Vandenberghe, 2004). In this method, instead of imposing the hard constraints implicit in the prior, we add a soft logarithmic barrier function to the log-likelihood. More precisely, we obtain an auxiliary estimate,  $\mathbf{x}_\epsilon$ , as

$$\mathbf{x}_\epsilon = \arg \max_{\mathbf{x}} \left\{ \log p(\mathbf{r}|\mathbf{x}) + \epsilon \sum_i [\log(c - x_i) + \log(c + x_i)] \right\}, \quad (10)$$

where the last sum is over all the components of  $\mathbf{x}$ . (We decoded one second of the movie here, so the stimulus dimensionality is  $120 \times 64 = 7680$ .) The barrier stiffness parameter,  $\epsilon$ , is iteratively reduced to zero in an outer loop, letting  $\mathbf{x}_\epsilon$  converge to the true solution of the constrained problem. The key to the efficiency of this method is that the Hessian of the logarithmic terms proportional to  $\epsilon$  in equation (10) is diagonal. Therefore, the total Hessian of the modified objective function remains banded (since as explained in Sec. 3.1,



the Hessian of the log-likelihood is banded), and the Newton-Raphson method for finding  $\mathbf{x}_c$  can be found in  $O(T)$  computational time. (See (Koyama and Paninski, 2009; Paninski et al., 2010) for further discussion.)

In this case, after finding the MAP,  $\mathbf{x}_{\text{map}}$ , with the flat prior, we truncated each component of  $\mathbf{x}_{\text{map}}$  to the closer value in  $\{-c, c\}$  to obtain an estimate with binary values. Figure 5 shows the true and decoded stimuli in rasterized form; these stimuli can also be seen in movie form in the supplemental materials. Note that the decoded movie displays spatiotemporal correlations which are absent in the original white noise movie; this illustrates the fact that the covariance of the conditional stimulus distribution  $p(\mathbf{x}|\mathbf{r})$  is often significantly different from that of the prior  $p(\mathbf{x})$ . Also note that much of the high-frequency detail of the movie is lost, due again to the lowpass characteristics of the stimulus filter matrix  $K$ ; c.f. Fig. 2e.

In Fig. 6, we examine the effect of using a Gaussian surrogate prior instead of a flat prior, as described above. We performed MAP decoding under a Gaussian surrogate prior, with mean and variance set to match the first two moments of the flat prior. Decoding under this prior was performed using the  $O(T)$  unconstrained method described in section 3.1. The Gaussian prior has a constant curvature in the log-domain; thus, in regions where information carried by the spike trains regarding the true stimulus is low (e.g., due to low firing rate), this decoder shrinks the estimate towards the prior mean (zero). The uniform prior, on the other hand, is flat (completely uninformative) away from the boundaries  $\pm c$ , and therefore the constrained decoder has a tendency to “stick” to either  $c$  or  $-c$ , as weak information from the likelihood term  $p(\mathbf{r}|\mathbf{x})$  shifts the gradient of the log-posterior slightly positive or negative, respectively. See (Ahmadian et al., 2010) for further discussion of this effect.

Figure 7 shows an example of MAP estimation applied to a high-dimensional spatial stimulus. We presented a 1,024-dimensional stimulus (a 32x32 image, shown in 7A) to a set of 1,024 simulated neurons (512 ON and 512 OFF cells, with center-surround receptive fields arranged in complementary square lattices tiling the image plane). Here we once again compared MAP decoding performance under an independent Gaussian prior and a correlated Gaussian prior with  $1/F^2$  scaling of spatial frequency components, a commonly used description of the power spectrum of natural images. Decoding under the correlated prior is significantly better than under an independent prior, even though a  $1/F^2$  Gaussian provides a relatively impoverished model of natural image statistics (Simoncelli, 2005). This suggests that significant improvements in decoding of naturalistic stimuli could be obtained by coupling a more powerful model of image statistics to an accurate neural encoding model under the Bayesian framework.

We also performed decoding of a much higher-dimensional 128x128 stimulus, using the responses of 1,024 cells (Fig. 7B). Estimating  $\mathbf{x}_{\text{map}}$  in the full stimulus space here is computationally prohibitive; for instance, the Hessian has  $128^4 > 10^8$  elements. However, in some cases we can proceed by working within a subspace determined by the prior covariance and the receptive fields of the neurons recorded. We begin by assuming that the log-prior may be written in the form

$$\log p(\mathbf{x}) = q(\mathbf{x}^T \mathbf{C}^{-1} \mathbf{x}), \quad (11)$$

with  $\mathbf{C}^{-1}$  a positive definite matrix; such a log-prior can specify any concave, elliptically-symmetric function of  $\mathbf{x}$  (Lyu and Simoncelli, 2009). Similarly, note that the log-likelihood may be written as  $\log p(\mathbf{r}|\mathbf{x}) = w(K\mathbf{x})$  for a suitable function  $w(\cdot)$ , emphasizing that this likelihood term only depends on  $\mathbf{x}$  through a projection onto a subspace spanned by the columns of the filter matrix  $K$  (a short, fat matrix, if [# stimulus dimensions] > [#

neurons]). Thus we write the log-posterior as

$$\log p(\mathbf{x}|\mathbf{r}) = q(\mathbf{x}^T \mathcal{C}^{-1} \mathbf{x}) + w(K\mathbf{x}). \quad (12)$$

To maximize this function it turns out we may restrict our attention to a subspace; that is, we do not need to perform a search over the full high-dimensional  $\mathbf{x}$ . To see this, we use a linear change of variables,

$$\mathbf{y} = A^{-1} \mathbf{x},$$

where

$$AA^T = \mathcal{C}.$$

This allows us to rewrite the log-posterior as

$$q(\mathbf{y}^T \mathbf{y}) + w(KA\mathbf{y}). \quad (13)$$

Now, by the log-concavity of the prior  $p(\mathbf{x})$ , the function  $q(u)$  must be a non-increasing function of  $u > 0$ . This implies, by the representer theorem (Scholkopf and Smola, 2002), that we may always choose an optimal  $\mathbf{y}$  in the space spanned by the rows of  $KA$ ; this is because increasing  $\mathbf{y}$  in a direction orthogonal to  $KA$  does not change the second term in the log-posterior but cannot increase the first term (since  $q(\cdot)$  is non-increasing).

So we may perform our optimization in the lower-dimensional subspace spanned by the rows of  $KA$ . If  $B$  is a matrix whose columns form an orthonormal basis for the row space

of  $KA$ , we can rewrite the log-posterior as

$$q(\vec{z}^T \vec{z}) + w(KAB\vec{z}), \quad (14)$$

where  $\vec{z}$  is now a vector of dimensionality equal to the rank of  $KA$ . In Fig. 7B, this reduces the dimensionality of the search from  $128^2 = 16,384$  to a much more feasible  $1,024$ .<sup>4</sup> Once an optimal  $\mathbf{z}_{\text{map}}$  is computed, we need only set  $\mathbf{x}_{\text{map}} = AB\mathbf{z}_{\text{map}}$ .

The remaining problem is to compute the change-of-basis operator  $A$  satisfying  $AA^T = \mathcal{C}$ . Naively, we could compute  $A$  via a Cholesky decomposition of  $\mathcal{C}$ , but this may be computationally infeasible given that  $A$  and  $\mathcal{C}$  are both  $d \times d$  matrices, where  $d$  is the stimulus dimensionality (here 16,384). The problem can be solved much more easily in cases where we have a known diagonalization of the prior covariance  $\mathcal{C} = O^T D O$ , where  $O$  is an orthogonal matrix and  $D$  diagonal.

For example, in the present case of a  $1/F^2$  Gaussian prior,  $O$  and  $O^T$  correspond to the 2D Fourier and inverse-Fourier transform, respectively, and  $D$  is a diagonal matrix with the weights  $1/F^2$  (where  $F$  is the frequency of the corresponding Fourier component). Thus we can use  $A = O D^{1/2} O^T$ , which is easy to compute since  $D$  is diagonal. Moreover, we can compute  $KA$  without explicitly representing  $A$ , since we can use the fast-Fourier transform in place of multiplication by  $O$  (and inverse transform in place of multiplication by  $O^T$ ). Thus we may in some cases tractably perform MAP decoding even for very high-dimensional spatial stimuli  $\mathbf{x}$ . Note, however, that this trick is useful only in the case that we have a stimulus with higher dimensionality than the number of recorded neurons.

---

<sup>4</sup>Note that in general, this trick is only useful in the case that  $N \times T$  (the number of neurons  $N$  times the number of time points in the linear filter  $T$  — i.e., the dimensionality of the range space of  $K$ ) is significantly less than the number of stimulus dimensions  $d$ .

## 4 Connections between the MAP and the OLE

Several important connections exist between the MAP estimate and the optimal linear estimator (OLE) (Bialek and Zee, 1990; Rieke et al., 1997). To explore these connections, it is helpful to begin by examining a slightly simpler model, where the MAP and the OLE coincide exactly. Assume for a moment the following Gaussian model for the spike train responses  $\mathbf{r}$ :

$$r_i \sim \mathcal{N}((K\mathbf{x})_i + b, \sigma^2); \quad \mathbf{x} \sim \mathcal{N}(0, \mathcal{C}),$$

where  $(K\mathbf{x})_i$  denotes the  $i$ 'th element of the vector  $K\mathbf{x}$ . Here, the observed responses are Gaussian, with some baseline  $b$ . The filter matrix  $K$  controls the dependence of the responses  $r_i$  on the stimulus  $\mathbf{x}$ ; as above,  $K$  acts as a convolution matrix corresponding to the linear filter  $\mathbf{k}$  in the GLM. In this case it is easy to see that the MAP is exactly the OLE, since the posterior distribution  $p(\mathbf{x}|\mathbf{r})$  is Gaussian, with covariance independent of the observed  $\mathbf{r}$ . In particular, in this case the OLE and the MAP are both given by

$$\mathbf{x}_{\text{ole}} = \mathbf{x}_{\text{map}} = (\sigma^2\mathcal{C}^{-1} + K^TK)^{-1}K^T(\mathbf{r} - b). \quad (15)$$

This solution has several important and intuitive features. Let  $\|\mathbf{k}\|$  denote the norm of the filter  $\mathbf{k}$ , and  $c$  denote the stimulus contrast, so that  $\mathcal{C} \propto c^2$ . First, in the low signal-to-noise regime (that is, for high values of  $\sigma$ , or equivalently small values of  $c\|\mathbf{k}\|$ ), the solution looks like  $(1/\sigma^2)\mathcal{C}K^T(\mathbf{r} - b)$ , which is just the convolution of  $\mathbf{r}$  with  $\mathbf{k}$  in the case that the stimulus covariance  $\mathcal{C}$  is white. Conversely, in the high SNR regime, where  $c\|\mathbf{k}\|$  is much larger than  $\sigma$ , the solution looks more like  $(K^TK)^{-1}K^T(\mathbf{r} - b)$ , where  $(K^TK)^{-1}K^T$  is the Moore-Penrose pseudo-inverse of  $K$ .<sup>5</sup> Thus, in the high-SNR limit the MAP (and the

---

<sup>5</sup>Assuming, for simplicity, that  $K^TK$  has full rank.

OLE) effectively *deconvolves*  $\mathbf{r}$  by  $\mathbf{k}$ . Note in addition that, as expected, the effect of the prior covariance  $\mathcal{C}$  disappears in the high-SNR limit, where the likelihood term dominates the prior term.

We can show that the same basic behavior (convolution in the low-SNR regime and deconvolution in the high-SNR regime) holds in the GLM, and that the MAP and OLE—although not equivalent in general—coincide exactly in the low-SNR limit. We provide a full derivation of this result in Appendix A, but we sketch the primary results here. The OLE is defined as

$$\mathbf{x}_{\text{ole}} = (\mathbb{E}[\mathbf{r}_0 \mathbf{r}_0^T])^{-1} \mathbb{E}[\mathbf{r}_0 \mathbf{x}^T]^T \mathbf{r}_0, \quad (16)$$

where  $\mathbb{E}[\cdot]$  denotes expectation over the joint density  $p(\mathbf{r}, \mathbf{x})$ ,  $\mathbf{r}_0$  are the mean-subtracted responses  $\mathbf{r}_0 = \mathbf{r} - \mathbb{E}[\mathbf{r}]$ , and (as above) the stimulus is distributed as  $\mathbf{x} \sim \mathcal{N}(0, \mathcal{C})$ .

For the OLE based on spike trains generated by the GLM, we need to compute  $\mathbb{E}[\mathbf{r}_0 \mathbf{r}_0^T]$ , the auto-covariance of the response, and  $\mathbb{E}[\mathbf{r}_0 \mathbf{x}^T]$ , the covariance of the response with the stimulus (which is equivalent to the “spike-triggered average”). We show that in this case,

$$\mathbf{x}_{\text{ole}} = \mathcal{C} K^T \text{diag}[f'(\mathbf{b}) ./ f(\mathbf{b})] (\mathbf{r} - dt f(\mathbf{b})) + o(c \|\mathbf{k}\|) = \mathbf{x}_{\text{map}} + o(c \|\mathbf{k}\|), \quad (17)$$

which holds exactly in the low-SNR regime  $c \|\mathbf{k}\| \rightarrow 0$ ; here  $\mathbf{a} ./ \mathbf{b}$  denotes the pointwise quotient of the vectors  $\mathbf{a}$  and  $\mathbf{b}$ . Once again, for larger SNR, the MAP displays pseudo-inverse-like behavior; see Appendix A for details.

In Figure 8, we show an explicit comparison of MAP and OLE decoding performance as a function of contrast and the number of cells in the population, for spike trains generated by a GLM (Fig. 1). As expected from our derivation, the errors in the OLE and MAP estimates converge at low contrast (i.e., low SNR). However, the MAP significantly

outperforms the OLE at high contrast. The MAP estimator also outperforms the OLE when employed with large numbers of cells, which corresponds to increasing the effective SNR.

In the case of a GLM with an exponential nonlinearity,  $f(\cdot) = \exp(\cdot)$ , the above formula can be simplified in a way that provides insight into the decoding significance of model components such as a spike-history dependent term. Specifically, we have

$$\mathbf{x}_{\text{map}} = \mathcal{C}K^T (\mathbf{r} - dt \exp(\mathbf{b} + B\mathbf{r})) + o(\|K\|), \quad (18)$$

where  $B$  is a linear operator capturing the causal dependence of the response on spike-train history. Thus we have the rather intuitive result that spike history effects (to first order in  $K$ ) simply weight the baseline firing rate in the MAP estimate; see Fig. 9 for an illustration of this effect.

## 5 Computing information-theoretic quantities

A number of previous authors have drawn attention to the connections between the decoding problem and the problem of estimating how much information (in the Shannon sense (Cover and Thomas, 1991)) a population spike train carries about a stimulus (Bialek et al., 1991; Rieke et al., 1997; Warland et al., 1997; Barbieri et al., 2004). In general, estimating this mutual information is quite difficult, particularly in high-dimensional spaces (Paninski, 2003). But in the case that our forward model of  $p(\mathbf{r}|\mathbf{x})$  is sufficiently accurate, the methods discussed here permit tractable computation of the mutual information. (See, e.g., (Nemenman et al., 2004; Kennel et al., 2005) for alternate approaches towards estimating the information which are model-based, but in a more nonparametric sense than

the methods developed here.)

We can write the mutual information (MI) as

$$\begin{aligned} I[\mathbf{x}; \mathbf{r}] &= H[\mathbf{x}] - H[\mathbf{x}|\mathbf{r}] \\ &= H[\mathbf{x}] - \int p(\mathbf{r}) \left( - \int p(\mathbf{x}|\mathbf{r}) \log p(\mathbf{x}|\mathbf{r}) d\mathbf{x} \right) d\mathbf{r}. \end{aligned}$$

The first term, the prior stimulus entropy, depends only on the prior stimulus distribution  $p(\mathbf{x})$ , which in the case of artificial stimuli is set by the experimenter (and whose entropy may typically therefore be computed exactly). In the case of natural stimulus ensembles we can draw an arbitrarily large number of samples from  $p(\mathbf{x})$ , and may therefore in principle still consider computing  $H[\mathbf{x}]$  to be a solvable problem.

The second term, which is sometimes referred to as the “noise entropy” (Strong et al., 1998), is the average residual entropy in  $\mathbf{x}$  conditional on the spike data  $\mathbf{r}$ . Although residual entropy is generally much more difficult to compute, in the case that our Gaussian approximation to the posterior is acceptably accurate, we can apply a simple shortcut by using the well-known formula for the entropy of a Gaussian. Namely, for any specific instantiation of the observed spike data  $\mathbf{r}$ , we have

$$- \int p(\mathbf{x}|\mathbf{r}) \log p(\mathbf{x}|\mathbf{r}) d\mathbf{x} \approx -\frac{1}{2} \log |J| + \frac{d}{2} \log(2\pi e), \quad (19)$$

where we have used the formula for the entropy of a Gaussian distribution with covariance matrix  $J^{-1}$  (Cover and Thomas, 1991). As above,  $J$  denotes the Hessian of the negative log-posterior computed at  $\mathbf{x}_{\text{map}}$ . (Similar Fisher information-based approximations to the Shannon information have been discussed in many other contexts, e.g. (Clarke and Barron, 1990; Brunel and Nadal, 1998).) We need only average this entropy over the



data distribution  $p(\mathbf{r}) = \int p(\mathbf{x})p(\mathbf{r}|\mathbf{x})d\mathbf{x}$ . This averaging may be performed most easily via standard Monte Carlo numerical integration techniques (Press et al., 1992; Robert and Casella, 2005), that is, averaging this posterior-based entropy over many stimulus-response pairs (with stimulus drawn from the prior and response generated by the neural population).

Thus, to summarize, computing this approximation to the information  $I[\mathbf{x}; \mathbf{r}]$  requires that we:

1. Draw i.i.d. samples  $\mathbf{x}_j$  from the stimulus distribution  $p(\mathbf{x})$
2. Draw sample spike trains  $\mathbf{r}_j$  from the corresponding conditional distributions  $p(\mathbf{r}|\mathbf{x}_j)$ , either by observing spike responses from a real neuron or sampling spike responses from the point-process model
3. Compute the MAP estimate  $\mathbf{x}_{\text{map}}(\mathbf{r}_j)$  and Hessian  $J(\mathbf{r}_j)$  corresponding to the observed data  $\mathbf{r}_j$
4. Compute the approximate posterior entropy (eq. 19)
5. Form the average over all of our Monte Carlo samples:

$$\begin{aligned}
 H[\mathbf{x}|\mathbf{r}] &= \int p(\mathbf{r}) \left( - \int p(\mathbf{x}|\mathbf{r}) \log p(\mathbf{x}|\mathbf{r}) d\mathbf{x} \right) d\mathbf{r} \\
 &\approx \int p(\mathbf{r}) \left( - \frac{1}{2} \log |J(\mathbf{r})| + \frac{d}{2} \log(2\pi e) \right) d\mathbf{r} \\
 &= \left( \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{j=1}^N -\frac{1}{2} \log |J(\mathbf{r}_j)| \right) + \frac{d}{2} \log(2\pi e) = \hat{H}[\mathbf{x}|\mathbf{r}] \quad (20)
 \end{aligned}$$

6. Finally, subtract the result from the prior entropy:

$$I[\mathbf{x}; \mathbf{r}] \approx H[\mathbf{x}] - \hat{H}[\mathbf{x}|\mathbf{r}].$$

In practice, of course, the number of Monte Carlo samples  $N$  does not have to tend to infinity, but merely has to be large enough to make the confidence interval on the empirical average acceptably small. This method should give accurate estimates of the information whenever the posterior  $p(\mathbf{x}|\mathbf{r})$  may be well-approximated by a Gaussian, as is the case for the examples analyzed here (c.f. Fig. 3).

We can compare this estimate with a well-known lower-bound on  $I[\mathbf{x}; \mathbf{r}]$  which arises from a Gaussian approximation to the residuals obtained under linear decoding (Bialek et al., 1991; Rieke et al., 1997). This bound may be derived as follows: we use the data processing inequality

$$H[\mathbf{x}] - H[\mathbf{x}|\mathbf{r}] = I[\mathbf{x}; \mathbf{r}] \geq I[\mathbf{x}; \hat{\mathbf{x}}(\mathbf{r})] = H[\mathbf{x}] - H[\mathbf{x}|\hat{\mathbf{x}}],$$

where  $\hat{\mathbf{x}}$  is any estimator of  $\mathbf{x}$  given  $\mathbf{r}$ , to establish that

$$H[\mathbf{x}|\mathbf{r}] \leq H[\mathbf{x}|\hat{\mathbf{x}}].$$

If we write out this conditional entropy, we see that

$$\begin{aligned} H[\mathbf{x}|\hat{\mathbf{x}}] &= \mathbb{E}_{\mathbf{r}} \left[ - \int p(\mathbf{x}|\hat{\mathbf{x}}_{\mathbf{r}}) \log p(\mathbf{x}|\hat{\mathbf{x}}_{\mathbf{r}}) d\mathbf{x} \right] \\ &\leq \mathbb{E}_{\mathbf{r}} \left[ \frac{1}{2} \log |\text{cov}(\mathbf{x}|\hat{\mathbf{x}}_{\mathbf{r}})| + \frac{d}{2} \log(2\pi e) \right] \end{aligned} \tag{21}$$

$$\leq \frac{1}{2} \log \left| \mathbb{E}_{\mathbf{r}} \left[ \text{cov}(\mathbf{x}|\hat{\mathbf{x}}_{\mathbf{r}}) \right] \right| + \frac{d}{2} \log(2\pi e), \tag{22}$$

where  $\mathbb{E}_{\mathbf{r}}$  denotes expectation with respect to  $p(\mathbf{r})$ . The first inequality follows from the fact that a Gaussian has the highest entropy among all densities with a given covariance (Cover and Thomas, 1991), and the second inequality is Jensen's (since the log-determinant is a concave function).

This upper bound on  $H[\mathbf{x}|\mathbf{r}]$  provides us with a lower bound on  $I[\mathbf{x}; \mathbf{r}]$ . We can estimate the last line of this bound numerically by drawing many stimulus-response pairs  $\{\mathbf{x}_j, \mathbf{r}_j\}$ , computing residuals of the estimator  $\hat{\mathbf{x}}$ ,<sup>6</sup> given by  $\chi_j = \mathbf{x}_j - \hat{\mathbf{x}}$ , and then computing the covariance of these residuals,  $\mathbb{E}[\chi_j \chi_j^T]$ . We thus have

$$I[\mathbf{x}; \mathbf{r}] \geq H[\mathbf{x}] - \left( \frac{1}{2} \log \left| \mathbb{E}[\chi_j \chi_j^T] \right| + \frac{d}{2} \log(2\pi e) \right). \quad (23)$$

Figure 10 shows a comparison of the lower bound obtained by this method with an estimate of  $I[\mathbf{x}|\mathbf{r}]$  obtained directly from this Gaussian approximation to the posterior. For completeness, we also compare to the lower bound obtained by using  $\mathbf{x}_{\text{map}}$  instead of  $\mathbf{x}_{\text{ole}}$  in the above derivation (this second lower bound is guaranteed to be tighter if the MAP residuals are smaller than those of the OLE, as we observed in Fig. 8). For this example, using the responses from 32 neurons stimulated a 60-sample Gaussian white noise stimulus, the lower bound obtained from the MAP residuals is closely matched to the Laplace approximation-based estimate. However, the latter estimate converges much more rapidly to the true information, and is free of bias even for small numbers of stimulus-response pairs. This bias (middle panel, Fig. 10) arises from the fact that the residual-based estimate is not actually a lower bound unless the estimated covariance of the residuals has adequately converged; for small numbers of stimulus-response pairs, the residual covariance matrix is under-sampled, leading to an under-estimate of the residual entropy (and an over-estimate of MI). The Laplace-approximation based estimate of information is therefore an accurate and data-efficient alternative to the lower-bound estimates based on the OLE or MAP residuals.

The MAP-based lower-bound on mutual information appears to be relatively accurate

---

<sup>6</sup>In most applications this estimator  $\hat{\mathbf{x}}$  is taken as the OLE, but any estimator which is a function of the data  $\mathbf{r}$  may be employed here.

for a population of neurons with response properties matched to those in primate retina. However, we compared MI estimators using a second neural population with higher gain (i.e., large  $\|\mathbf{k}\|$ ) and longer refractory periods, and found a significant gap between the Laplace approximation-based estimate and the MAP-based lower bound, indicating that the lower bound may be loose for populations with highly nonlinear encoding (Fig. 11).

For insight into the gap between the lower bounds and our estimate (both of which rely on some form of Gaussian approximation), observe that the lower bounds characterize all the residuals as arising from a single Gaussian with fixed covariance. This ignores data-dependent variations in the width of the posterior along different feature axes (i.e. differing levels of uncertainty about particular stimulus features, which may arise due to the particular stimulus drawn from the prior, or stochasticity in spike response generated on individual trials). Thus, when the response nonlinearity induces response-dependent fluctuations in the Hessian of the log-posterior (as the GLM does at high contrasts), we can expect the OLE and MAP-based lower bounds to significantly underestimate the true mutual information between stimulus and response. This is the content of inequality (22).

We can quantify the contribution of such fluctuations by forming a third estimator (denoted  $\langle \text{Laplace} \rangle$  in Fig. 11), which uses the average Hessian  $\langle J \rangle = \frac{1}{N} \sum_j J(\mathbf{r}_j)$  to estimate  $H[\mathbf{x}|\mathbf{r}]$  instead of averaging  $\log |J(\mathbf{r}_j)|$  across responses (see eq. 20). For this estimator,

$$\hat{H}[\mathbf{x}|\mathbf{r}] = -\frac{1}{2} \log |\langle J \rangle| + \frac{d}{2} \log(2\pi e). \quad (24)$$

By transposing the average over  $J(\mathbf{r}_j)$  and the negative log-determinant, this estimate forms an upper bound on the Laplace-based conditional entropy estimate (eq. 20), by Jensen’s inequality, and thus a lower bound on the Laplace-based M.I. estimate. As shown in Fig. 11, this estimator accounts for much of the gap between the MAP-based lower

bound (eq. 23) and the Laplace-based estimate, showing that trial-to-trial fluctuations in the posterior can cause the MAP-based lower bound to underestimate the M.I. at high contrasts.

## 6 Discrimination and detection; change-point analysis

In the above we have been discussing estimation of continuous-valued parameters. However, it is important to note that similar methods are quite useful for two-point discrimination (aka detection) problems as well. Consider the following two-alternative forced choice (2AFC) paradigm: we observe a spike train, or population spike train,  $\mathbf{r}$ , and are asked to discriminate between two possible known stimuli,  $\mathbf{x}_0$  and  $\mathbf{x}_1$ , which might have produced the observed responses. In the statistics literature this 2AFC paradigm is known as testing between two simple hypotheses, and the optimal discriminator is known to be based on the posterior ratio  $p(\mathbf{x}_0|\mathbf{r})/p(\mathbf{x}_1|\mathbf{r})$ : if this ratio is greater than some threshold value, we say that  $\mathbf{x}_0$  was the stimulus, and otherwise we say it was  $\mathbf{x}_1$  (Schervish, 1995). See, e.g., (Pillow et al., 2008) for a recent application to retinal data, or (de Ruyter van Steveninck and Bialek, 1995; Rieke et al., 1997) for a nice list of applications of this idea in the classical psychophysics and neuroethology literature.

Now let us consider a slightly more general and realistic case, in which neither  $\mathbf{x}_0$  nor  $\mathbf{x}_1$  are known exactly. We have two hypotheses,  $H_0$  and  $H_1$ , and stimuli are drawn according to two distinct distributions  $p(\mathbf{x}|H_0)$  and  $p(\mathbf{x}|H_1)$ . Our goal is to decide which of the two distributions the stimulus was drawn from, given not the stimulus but just spiking data  $\mathbf{r}$ . (We will discuss more concrete examples below, but for now it may be helpful to keep the

following simple example in mind:  $\mathbf{x}$  is a white Gaussian noise stimulus, with one mean and variance under  $H_0$  and a different mean and variance under  $H_1$ ; our goal is to decide between these two hypotheses. Of course, more complex distributions  $p(\mathbf{x}|H_0)$  and  $p(\mathbf{x}|H_1)$  are feasible.)

In this case the optimal decision rule still has the form of a posterior-ratio test

$$\frac{p(H_0|\mathbf{r})}{p(H_1|\mathbf{r})} = \frac{p(\mathbf{r}|H_0)p(H_0)}{p(\mathbf{r}|H_1)p(H_1)} = \frac{p(H_0) \int p(\mathbf{r}|\mathbf{x})p(\mathbf{x}|H_0)d\mathbf{x}}{p(H_1) \int p(\mathbf{r}|\mathbf{x})p(\mathbf{x}|H_1)d\mathbf{x}}. \quad (25)$$

Thus we need to marginalize out the stimulus  $\mathbf{x}$ , which is not observed directly, to calculate  $p(\mathbf{r}|H)$ . (This ratio of marginal probabilities is called the ‘‘Bayes factor’’ in the Bayesian hypothesis testing literature (Kass and Raftery, 1995)). The key point is that we can directly adapt the Gaussian approximation described above to compute these integrals.

As before (eq. 9), we approximate the posterior by a Gaussian

$$\begin{aligned} G_1(\mathbf{x}) &= \frac{1}{z_1} p(\mathbf{r}|\mathbf{x})p(\mathbf{x}|H_1) \\ G_2(\mathbf{x}) &= \frac{1}{z_2} p(\mathbf{r}|\mathbf{x})p(\mathbf{x}|H_2), \end{aligned} \quad (26)$$

where  $G_i(\mathbf{x})$  is the Gaussian with mean  $\mathbf{x}_{\text{map}_i}$  and covariance  $C_i = J_i^{-1}$  (both computed using the prior  $p(\mathbf{x}|H_i)$ ), and  $z_i$  is an unknown normalization constant for each posterior (known in the Bayesian statistical literature as the ‘‘evidence’’ or ‘‘marginal likelihood’’ (Kass and Raftery, 1995)). Now clearly

$$\int p(\mathbf{r}|\mathbf{x})p(\mathbf{x}|H_i)d\mathbf{x} = \int z_i G_i(\mathbf{x})d\mathbf{x} = z_i, \quad (27)$$

so to compute our posterior ratio (eq. 25) we just need to compute  $z_1$  and  $z_2$ .

From the encoding model, we know the value  $p(\mathbf{r}|\mathbf{x}_{\mathbf{map}_i})$  and from our prior on  $\mathbf{x}$  we know  $p(\mathbf{x}_{\mathbf{map}_i}|H_i)$ . We also know from the formula for a Gaussian density that

$$G_i(\mathbf{x}_{\mathbf{map}_i}) = [(2\pi)^{d/2}|C_i|^{1/2}]^{-1} = |J_i|^{1/2}/(2\pi)^{d/2}.$$

So by inserting these terms into (eq. 26) and solving for  $z_i$  we obtain

$$\begin{aligned} z_i &= \frac{p(\mathbf{r}|\mathbf{x}_{\mathbf{map}_i})p(\mathbf{x}_{\mathbf{map}_i}|H_i)}{G(\mathbf{x}_{\mathbf{map}_i})} \\ &= p(\mathbf{r}|\mathbf{x}_{\mathbf{map}_i})p(\mathbf{x}_{\mathbf{map}_i}|H_i)(2\pi)^{d/2}|C_i|^{1/2}, \end{aligned}$$

and the Bayes factor reduces to

$$\frac{p(H_0|\mathbf{r})}{p(H_1|\mathbf{r})} = \frac{p(H_0)z_0}{p(H_1)z_1} = \frac{p(H_0)p(\mathbf{r}|\mathbf{x}_{\mathbf{map}_0})p(\mathbf{x}_{\mathbf{map}_0}|H_0)|C_0|^{1/2}}{p(H_1)p(\mathbf{r}|\mathbf{x}_{\mathbf{map}_1})p(\mathbf{x}_{\mathbf{map}_1}|H_1)|C_1|^{1/2}}. \quad (28)$$

Thus, once again, the computation of these marginal posterior quantities reduces to a simple determinant computation once we have obtained  $\mathbf{x}_{\mathbf{map}}$  and  $J$  under each hypothesis, assuming the Gaussian approximation is accurate. The computation of the determinants  $|C_0|$  and  $|C_1|$  can be performed in  $O(T)$  time in many important cases; again, see (Paninski et al., 2010) for details. If this Gaussian approximation is inaccurate, on the other hand, more expensive Monte Carlo approaches are required; see (Ahmadian et al., 2010) for further details.

## 6.1 Optimal change-point detection

A more subtle and perhaps more behaviorally relevant situation arises when we are asked to detect the time at which the stimulus undergoes a change between class  $H_0$  to class

$H_1$  (for example, the time at which the mean or the variance of the stimulus is changed suddenly (DeWeese and Zador, 1998)). We may compute the posterior probability of “no change” exactly as before, using our Gaussian approximation-based estimate of  $p(\mathbf{r}|H_0)$ . Now the likelihood that a change occurred at time  $t$  is

$$p(\mathbf{r}|H_{[\text{change at } t]}) = \int p(\mathbf{r}|\mathbf{x})p(\mathbf{x}|H_{[\text{change at } t]})d\mathbf{x}.$$

Thus finding the time at which the change occurs simply requires that we compute

$$\begin{aligned} p(\mathbf{r}|H_{[\text{change at } t]}) &= \int p(\mathbf{r}|\mathbf{x})p(\mathbf{x}|H_{[\text{change at } t]})d\mathbf{x} \\ &\approx p(\mathbf{r}|\mathbf{x}_{\text{map}})p(\mathbf{x}_{\text{map}}|H_{[\text{change at } t]})(2\pi)^{d/2}|C_t|^{1/2}, \end{aligned} \quad (29)$$

where, again, the MAP estimate and approximate covariance  $C_t$  for each time  $t$  are computed under the prior distribution  $p(\mathbf{x}|H_{[\text{change at } t]})$ .

Choosing the peak of this function gives us the maximum-likelihood estimator for the change-point time; on the other hand, the posterior probability that a change occurred at all is given by averaging:

$$p(\text{change at any time}) = \int p(\mathbf{r}|H_{[\text{change at } t]})p(t)dt,$$

with  $p(t)$  the experimentally-controlled prior distribution on change point times (which might, for example, be chosen to be uniform on some interval  $t \in (a, b)$ ). Figure 12 shows an example of the change-point detection task, illustrating detection of a change in mean (left) and a change in variance (right).



## 7 Discussion

We have described three techniques for model-based decoding of neural spike trains: (1) efficient computational methods for computing the MAP estimate, based on the GLM encoding model; (2) a tractable method for estimating the mutual information between the stimulus and the response; and (3) methods for change-point detection based on marginal likelihood. These three ideas are connected by a simple Gaussian approximation of the (log-concave) posterior  $p(\mathbf{x}|\mathbf{r})$ : the MAP decoder corresponds to the peak location of this Gaussian; our estimate of the mutual information corresponds to the width of this Gaussian relative to the width (i.e., entropy) of the prior distribution; finally, the marginal likelihood corresponds to the height of the (unnormalized) Gaussian approximation to  $p(\mathbf{x}, \mathbf{r})$ , relative to the height of the normalized Gaussian  $p(\mathbf{x}|\mathbf{r})$ . Below we discuss connections between these ideas and previous contributions to the neural decoding literature.

### 7.1 Decoding and the Gaussian approximation

MAP techniques for neural decoding have been previously applied in several contexts. The work closest to ours is the extended abstract by (Stanley and Boloori, 2001) (see also (Jacobs et al., 2006)), where MAP decoding was proposed as a relatively tractable alternative to the full posterior mean solution  $\mathbb{E}[\mathbf{x}|\mathbf{r}]$  (which requires a high-dimensional integration) for decoding the binned firing rate generated by a single cascade-type model cell with truncated Gaussian outputs and no spike-history effects. These authors emphasized the superiority of the MAP estimate (which incorporates an explicit model of how the responses are generated) versus the OLE (which does not incorporate such an explicit encoding model) in this case, but did not discuss methods for quantifying the uncertainty

in the estimates or incorporating spike history effects or simultaneous observations from more than one neuron. In addition, the log-likelihood of the model introduced in (Stanley and Boloori, 2001) does not appear to share the concavity properties enjoyed by the point process GLM; these concavity properties clearly play a central role in our development.

(Lazar and Pnevmatikakis, 2008) summarizes another closely related thread of work. These authors address the problem of decoding a stimulus from the spike trains of a population of noiseless integrate-and-fire (IF) neurons that are driven by a linearly-filtered version of the stimulus. The key idea is that each spike provides a single linear constraint on the stimulus; by combining enough of these linear constraints, the stimulus can be uniquely recovered. (When the number of spikes is less than the dimensionality of the stimulus, or if noise is present, a pseudoinverse solution may be employed; a similar approach was discussed in the context of parameter estimation in (Pillow and Simoncelli, 2003).) It is worth noting that this intersection-of-constraints approach is not equivalent to the MAP approach we have discussed here, because the former only uses information about the spike times (when the model neuron voltage is exactly equal to the spiking threshold, leading to a linear equality constraint), while useful information about the non-spiking times (when the voltage is below the threshold) is discarded. The latter information comprises a set of inequality constraints which may be easily included in the MAP approach (Koyama and Paninski, 2009; Paninski et al., 2010); discarding these inequality constraints can lead to suboptimal decoding performance. (Gerwinn et al., 2009) discusses these issues in more depth, in the context of a noisy IF model; as emphasized in footnote 3, all of the methods introduced in this paper can be applied directly to the IF model discussed in (Paninski et al., 2004b; Pillow et al., 2005), since this IF model shares the log-concavity properties of the generalized linear model we have focused on here. (See also (Paninski, 2004; Paninski et al., 2007) for further discussion of the close connections between the point-process GLM

and this type of linearly-filtered integrate-and-fire model.)

A seminal paper in the MAP decoding literature is (de Ruyter and Bialek, 1988): the idea in this paper was to directly sample the conditional distributions  $p(\mathbf{x}|\mathbf{r})$  for certain simple examples of the observed spiking data  $\mathbf{r}$  (e.g., the authors collected samples from the empirical distribution of  $\mathbf{x}$  given a single spike, or a spike doublet separated by an interspike interval of length  $\tau$ , or a spike triplet indexed by two ISIs  $\tau_1$  and  $\tau_2$ ). Then a Gaussian model was fit to these spike-, doublet-, and triplet-triggered ensembles; this is exactly comparable to our Gaussian approximation of the posterior distributions here, except that our approximation is based on matching the first and second derivatives of the model-based  $p(\mathbf{x}|\mathbf{r})$  at the MAP solution and the approximation in (de Ruyter and Bialek, 1988) is based on matching first and second moments to the empirical distribution. These authors also proposed a heuristic method for combining the information in separate triplets to obtain a kind of pseudo-MAP estimate of  $\mathbf{x}$  given the full spike train (this combination rule is based on an independence assumption that the authors emphasize is unjustified in general). The model-based MAP estimate and Gaussian approximation proposed here may therefore be considered a more principled way to knit together information from multiple spike events (and multiple spike trains), even in the presence of significant dependencies between spikes (i.e., spike-history effects). Finally, it is worth noting that in the GLM, short sequences of spikes are informative only about projections of the stimulus onto spaces of low dimensionality, as (de Ruyter and Bialek, 1988) observed in their data.

Another important and influential series of papers, by Brown and colleagues (Brown et al., 1998; Barbieri et al., 2004; Truccolo et al., 2005; Srinivasan et al., 2006), has made use of a generalization of the Kalman filter proposed by (Fahrmeir, 1992) to perform approximately-optimal decoding efficiently in cases where the joint distribution of the stimulus and response  $(\mathbf{x}, \mathbf{r})$  may be written in the form of a state-space model with a

low-dimensional state space. These techniques provide an approximate MAP estimate of the full high-dimensional signal  $\mathbf{x}$  (instead of the exact optimization over  $\mathbf{x}$  described here). However, the update step in the recursive filter does depend on a simpler (low-dimensional) exact maximization over the value of  $\mathbf{x}$  at each individual time  $t$ , followed by a low-dimensional Hessian-based Gaussian approximation that is exactly analogous to the high-dimensional Gaussian approximation for  $p(\mathbf{x}|\mathbf{r})$  discussed here; see (Fahrmeir, 1992; Brown et al., 1998) for full details. One important advantage of the direct optimization approach described here is that we may obtain the exact MAP estimate (instead of an approximation), in many cases with the same  $O(T)$  computational complexity, even in cases where the stimulus  $\mathbf{x}(t)$  can not be easily described in terms of a state-space model. This is particularly important in the high-dimensional vision decoding problems discussed here, for example. Constrained problems are also easier to handle using this direct optimization approach (e.g., as shown in Fig. 5). See (Paninski et al., 2010) for further discussion.

Finally, in section 4 we discussed some important connections between the OLE and MAP estimates; as we emphasized above, the MAP and OLE match exactly in the limit of low SNR (Fig. 8), though the MAP is superior at high SNR, assuming the model is correctly specified. Moreover, both the MAP and OLE show a crossover between convolution- and deconvolution-like behavior as the SNR increases. Similar points were made in the context of a simpler version of the GLM (using a similar perturbative analysis) by (Bialek and Zee, 1990); further discussion appears in (Rieke et al., 1997).

We close this section by noting a major direction for extending the applicability of the MAP decoding methods described here. One of the major strengths of the GLM encoding model (and related models (Paninski et al., 2004b; Paninski et al., 2005; Kulkarni and Paninski, 2007; Pillow, 2009)) is that we may very easily incorporate nonlinear terms into the model. That is, instead of restricting our attention to models of the form  $\lambda(t) =$

$f(b + \mathbf{k} \cdot \mathbf{x} + \sum_j h(t - t_j))$ , we may incorporate nonlinear terms  $\lambda(t) = f(b + \mathbf{k} \cdot \mathbf{x} + \sum_i z_i \mathcal{F}_i(\mathbf{x}) + \sum_j h(t - t_j))$  and estimate the weights  $z_i$  by concave maximum likelihood, just like the other model parameters  $(b, \mathbf{k}, h(\cdot))$ ; this greatly increases the flexibility and power of this model. (As emphasized in (Chichilnisky, 2001; Paninski, 2004), this generalized model simply corresponds to a relabeling of our stimulus  $\mathbf{x}$ ; incorporating nonlinear terms of this nature is a standard technique in multiple regression analysis (Duda and Hart, 1972; Sahani, 2000).) However, while this nonlinear model retains its concavity in the model parameters, unfortunately it loses the key concavity in  $\mathbf{x}$ , and the likelihood of  $\mathbf{x}$  is therefore prone to non-global local maxima<sup>7</sup>. Handling this issue constitutes an important avenue for future research.

## 7.2 Information estimation

Perhaps our most striking result is that the linear reconstruction lower bound on mutual information may be significantly loose, depending on the stimulus strength and the properties of the encoding model. This lower bound technique has been employed quite frequently since its introduction by (Bialek et al., 1991) (see (Rieke et al., 1997) for a partial list of applications), and therefore the improved estimate introduced here may have a significant impact on our understanding of the fidelity of the neural code in a variety of experimental preparations. Previous authors have emphasized the looseness of this lower bound in applications to several model preparations, e.g. the cat retina (Passaglia and Troy, 2004) and the electrosensory system of the weakly electric fish (Chacron, 2005).

---

<sup>7</sup>To be clear, models including these nonlinearities do not necessarily have local maxima. Our log-concavity conditions are in a sense “over-engineered” to guarantee that no local maxima exist: these conditions are sufficient, not necessary. The point is that we can no longer *guarantee* that we have found the global maximum in these nonlinear models; in this case simulated annealing or Metropolis-Hastings methods are typically required to explore the stimulus space more thoroughly, as we discuss further in section 7.3 below.

As with the MAP decoding idea, a number of variants of our information estimate have appeared previously in the literature. In particular, (de Ruyter and Bialek, 1988) computed the entropy of the spike-triggered Gaussian approximation discussed above, in order to quantify the informativeness of single spikes versus spike doublet and triplet events; again, our model-based techniques may be considered a generalization in that they allow us to compute the conditional entropy given a population spike train containing an arbitrarily large number of spikes. In addition, (Barbieri et al., 2004) used their recursive approximate point process filter to compute dynamic estimates of the conditional entropy of  $\mathbf{x}(t)$  given the available spiking data up to time  $t$ . Finally, the idea that we might use model-based approaches to computing the information  $I[\mathbf{x}; \mathbf{r}]$  (a problem which is otherwise quite difficult when considered nonparametrically (Paninski, 2003)) has appeared in several earlier works (Harris et al., 2003; Butts and Stanley, 2003; Sharpee et al., 2004; Paninski et al., 2004a; Pillow and Simoncelli, 2006).

### 7.3 Extensions: fully-Bayesian techniques

The most important extension of these methods is to adapt these techniques to employ more general “fully Bayesian” methods, in which we compute these integrals exactly by Monte Carlo techniques (Robert and Casella, 2005) instead of the computationally cheaper Gaussian approximation used here. This extension is important for three reasons: first, it is necessary to verify our MAP-based results (especially concerning the slackness of the reconstruction lower bound on the mutual information) using the exact posterior densities, instead of the Gaussian approximation. Second, the MAP estimator can have a large average error in cases in which the stimulus prior is too flat, and the likelihood term  $p(\mathbf{r}|\mathbf{x})$  poorly constrains our estimate of  $\mathbf{x}$  (e.g., uniformly-distributed  $\mathbf{x}$ ; c.f. Fig. 6); in

this case, we expect the posterior mean estimate  $\mathbb{E}[\mathbf{x}|\mathbf{r}]$  to be superior. Finally, fully Bayesian methods allow us to consider a wider variety of convex cost functions than does the MAP framework; this flexibility in the choice of cost function may be important for some decoding applications.

A large variety of computational methods have been developed and intensively studied for computing the necessary integrals. In cases where the tree decomposition holds (e.g., the state space models discussed above), sequential importance sampling methods (particle filtering) can be quite effective (Doucet et al., 2001; Brockwell et al., 2004; Kelly and Lee, 2004; Shoham et al., 2005; Ergun et al., 2007; Brockwell et al., 2007). More generally, methods based on the Metropolis-Hastings algorithm (Robert and Casella, 2005) may be applied (Rigat et al., 2006; Cronin et al., 2009); we discuss these fully Bayesian methods in much more depth in the companion paper (Ahmadian et al., 2010).

## **Appendix A: Comparing the MAP estimator and OLE in the low-SNR regime**

As noted in Sec. 4, the behavior of the MAP estimator  $\mathbf{x}_{\text{map}}$  in the low signal-to-noise regime depends on the moments  $\mathbb{E}[\mathbf{r}_0\mathbf{x}^T]$  and  $\mathbb{E}[\mathbf{r}_0\mathbf{r}_0^T]$ . (Recall that  $\mathbf{r}_0$  is the mean-subtracted response,  $\mathbf{r}_0 = \mathbf{r} - \mathbb{E}[vr]$ ). In the limit of low-SNR  $c\|\mathbf{k}\| \rightarrow 0$ , these terms can be calculated as follows. Here, as in Sec. 4,  $\|\mathbf{k}\|$  denotes the norm of the filter  $\mathbf{k}$ , and  $c$  denotes the stimulus contrast, so that the magnitude of the stimulus  $\mathbf{x}$  is proportional to  $c$ ; in addition, for simplicity, we assume for now that the spike history terms  $h_{ij}(\cdot)$  are negligible for all  $(i, j)$ .

Under the GLM encoding model, responses are conditionally Poisson:

$$\mathbf{r} \sim \text{Pois}[f((K\mathbf{x}) + \mathbf{b})dt], \quad (30)$$

where  $f$  is the response nonlinearity, assume the stimulus has a Gaussian prior,  $\mathbf{x} \sim \mathcal{N}(0, \mathcal{C})$ . A second-order expansion in  $\mathbf{x}$  around  $\mathbf{b}$  gives

$$\mathbb{E}[\mathbf{r}|\mathbf{x}] = dt \left( f(\mathbf{b}) + f'(\mathbf{b}) \cdot K\mathbf{x} + \frac{1}{2} f''(\mathbf{b}) \cdot K\mathbf{x} \cdot K\mathbf{x} \right) + o(c^2 \|\mathbf{k}\|^2), \quad (31)$$

with ‘ $\cdot$ ’ again denoting pointwise multiplication of vectors. Averaging this over  $\mathbf{x}$ , we obtain

$$\mathbb{E}[\mathbf{r}] = dt \left( f(\mathbf{b}) + \frac{1}{2} f''(\mathbf{b}) \text{diag}[KCK^T] \right) + o(c^2 \|\mathbf{k}\|^2). \quad (32)$$

Given that  $\mathbf{x}$  has zero mean, we have  $\mathbb{E}[\mathbf{r}_0 \mathbf{x}^T] = \mathbb{E}[\mathbf{r} \mathbf{x}^T]$ , and since it has covariance  $\mathcal{C}$ , it follows that

$$\mathbb{E}[\mathbf{r}_0 \mathbf{x}^T] = \mathbb{E}[\mathbf{r} \mathbf{x}^T] = \mathbb{E}[\mathbb{E}[\mathbf{r}|\mathbf{x}] \mathbf{x}^T] = dt(\text{diag}[f'(\mathbf{b})]KC) + o(c^2 \|\mathbf{k}\|^2). \quad (33)$$

Now to derive  $\mathbb{E}[\mathbf{r}_0 \mathbf{r}_0^T]$ , note that

$$\mathbb{E}[\mathbf{r}_0 \mathbf{r}_0^T] = \mathbb{E}[\mathbb{E}[\mathbf{r}_0 \mathbf{r}_0^T | \mathbf{x}]] = \mathbb{E}[\text{Cov}[\mathbf{r}_0 | \mathbf{x}] + \mathbb{E}[\mathbf{r}_0 | \mathbf{x}] \mathbb{E}[\mathbf{r}_0^T | \mathbf{x}]]. \quad (34)$$

It follows from the Poisson assumption (eq. 30) that

$$\text{Cov}[\mathbf{r}_0 | \mathbf{x}] = \text{Cov}[\mathbf{r} | \mathbf{x}] = \text{diag}[\mathbb{E}[\mathbf{r} | \mathbf{x}]], \quad (35)$$



and from (eqs. 31–32) that

$$\mathbb{E}[\mathbf{r}_0|\mathbf{x}]\mathbb{E}[\mathbf{r}_0^T|\mathbf{x}] = dt^2(f'(\mathbf{b}).K\mathbf{x})(f'(\mathbf{b}).K\mathbf{x})^T + o(c^2\|\mathbf{k}\|^2), \quad (36)$$

so after averaging over  $\mathbf{x}$  we obtain

$$\begin{aligned} \mathbb{E}[\mathbf{r}_0\mathbf{r}_0^T] &= dt \left( \text{diag}[f(b)] + \frac{1}{2} \text{diag} [f''(\mathbf{b}).\text{diag}(K\mathcal{C}K^T)] \right. \\ &\quad \left. + dt \text{diag}[f'(\mathbf{b})]K\mathcal{C}K^T \text{diag}[f'(\mathbf{b})] \right) + o(c^2\|\mathbf{k}\|^2). \end{aligned} \quad (37)$$

Putting these pieces together to form the optimal linear estimator (eq. 16) gives

$$\mathbf{x}_{\text{ole}} = \mathcal{C}K^T \text{diag} [f'(\mathbf{b})./f(\mathbf{b})] (\mathbf{r} - dt(f(\mathbf{b}))) + o(c^2\|\mathbf{k}\|^2), \quad (38)$$

as discussed in Sec. 4. Thus we see that the OLE in the case of Poisson observations behaves much as in the case of Gaussian observations (eq. 15): in the low SNR regime, the OLE behaves like a convolution (with  $f'(\mathbf{b})./f(\mathbf{b})$  here playing the role of  $1/\sigma^2$  in the Gaussian case), while as the SNR increases the optimal linear filters take on the pseudoinverse form, with terms involving  $(K\mathcal{C}K^T)^{-1}$  multiplied by  $(\mathcal{C}K^T)$ .

Turning to the MAP case, we examine the log-posterior in a similar limit:

$$\log p(\mathbf{x}|\mathbf{r}) = g(\mathbf{b} + K\mathbf{x})^T \mathbf{r} - dt f(\mathbf{b} + K\mathbf{x})^T \mathbf{1} - \frac{1}{2} \mathbf{x}^T \mathcal{C}^{-1} \mathbf{x}, \quad (39)$$

where  $g(\cdot)$  abbreviates  $\log f(\cdot)$  and  $\mathbf{1}$  is a vector of all ones. Taking a second-order expansion

in  $\mathbf{x}$  gives

$$\begin{aligned} \log p(\mathbf{x}|\mathbf{r}) &= \left( g(\mathbf{b}) + g'(\mathbf{b}) \cdot K\mathbf{x} + \frac{1}{2} g''(\mathbf{b}) \cdot K\mathbf{x} \cdot K\mathbf{x} \right)^T \mathbf{r} - \frac{1}{2} \mathbf{x}^T \mathcal{C}^{-1} \mathbf{x} \\ &\quad - dt \left( f(\mathbf{b}) + f'(\mathbf{b}) \cdot K\mathbf{x} + \frac{1}{2} f''(\mathbf{b}) \cdot K\mathbf{x} \cdot K\mathbf{x} \right)^T \mathbf{1} + o(c^2 \|\mathbf{k}\|^2), \end{aligned} \quad (40)$$

This expression is quadratic in  $\mathbf{x}$ ; if we note that  $(f''(\mathbf{b}) \cdot K\mathbf{x} \cdot K\mathbf{x})^T \mathbf{1}$  and  $(g''(\mathbf{b}) \cdot K\mathbf{x} \cdot K\mathbf{x})^T \mathbf{r}$  may be written in the more standard form  $\mathbf{x}^T K^T \text{diag}[f''(\mathbf{b})] K\mathbf{x}$  and  $\mathbf{x}^T K^T \text{diag}[g''(\mathbf{b}) \cdot \mathbf{r}] K\mathbf{x}$ , respectively, then we may easily optimize to obtain

$$\begin{aligned} \mathbf{x}_{\text{map}} &= (\mathcal{C}^{-1} - K^T \text{diag}[\mathbf{r} \cdot g''(\mathbf{b}) - dt f''(\mathbf{b})] K)^{-1} K^T (g'(\mathbf{b}) \cdot \mathbf{r} - f'(\mathbf{b}) dt) + o(c^2 \|\mathbf{k}\|^2) \\ &= \mathcal{C} K^T \text{diag}[f'(\mathbf{b}) \cdot / f(\mathbf{b})] (\mathbf{r} - dt f(\mathbf{b})) + o(c \|\mathbf{k}\|). \end{aligned} \quad (41)$$

In the case that the nonlinearity  $f$  is exponential, the MAP takes a form that makes it easy to gain intuition about the effects of spike-history (and coupling) terms. Specifically, if the conditional intensity of the  $i$ 'th neuron is  $f((K\mathbf{x})_i + b_i + (B\mathbf{r})_i)$ , where  $B$  is a linear operator capturing the causal dependence of the response on spike-train history, then we obtain

$$\log p(\mathbf{x}|\mathbf{r}) = (\mathbf{b} + K\mathbf{x} + B\mathbf{r})^T \mathbf{r} - dt \exp(\mathbf{b} + K\mathbf{x} + B\mathbf{r})^T \mathbf{1} - \frac{1}{2} \mathbf{x}^T \mathcal{C}^{-1} \mathbf{x}; \quad (42)$$

optimizing to second order, as above, gives

$$\mathbf{x}_{\text{map}} = (\mathcal{C}^{-1} + dt K^T \text{diag}[\exp(\mathbf{b} + B\mathbf{r})] K)^{-1} K^T (\mathbf{r} - dt \exp(\mathbf{b} + B\mathbf{r})) + o(c^2 \|\mathbf{k}\|^2), \quad (43)$$

which, neglecting terms of second or higher order in  $c \|\mathbf{k}\|$ , reduces to (eq. 18) in the main text.

## Appendix B: GLM parameters and simulation details

Here we describe the parameters used for simulations and decoding analyses. For most analyses, the parameters of the GLM were those extracted from those fit to a single ON and a single OFF retinal ganglion cell from the population described in (Pillow et al., 2008) (shown here in Fig. 13A). We used an exponential nonlinearity to describe the mapping from linearly filtered input to conditional intensity, which provided a good description of retinal firing, i.e.,  $\lambda(t) = \exp(\mathbf{k} \cdot \mathbf{x}(t) + \sum_{\alpha} \mathbf{h}(t - t_{\alpha}) + b)$ . Each cell’s parameters therefore consisted of: (i) a stimulus filter  $\mathbf{k}$ ; (ii) a spike-history filter  $\mathbf{h}$ ; and (iii) a constant  $b$ . The stimulus filter  $\mathbf{k}$  was a purely temporal 40-tap filter, extracted as the first singular vector of the neuron’s full space-time receptive field estimated in (Pillow et al., 2008). The spike-history filter  $\mathbf{h}$  (Fig. 13A, right panel) was parametrized as a linear combination of 10 basis vectors ( “cosine bumps” ) of the form

$$\mathbf{b}_j(t) = \begin{cases} \frac{1}{2} \cos \left( \gamma \log \left[ \frac{t+\psi}{\phi_j+\psi} \right] \right) + \frac{1}{2}, & \text{if } \gamma \log \left[ \frac{t+\psi}{\phi_j+\psi} \right] \in [-\pi, \pi] \\ 0, & \text{otherwise,} \end{cases} \quad (44)$$

where  $\phi_j$  is the peak of the  $j$ ’th basis vector, and  $\gamma, \psi$  are scalars controlling the logarithmic stretching of time. We set the peaks of first and last basis vectors to  $\phi_1 = 1$  ms and  $\phi_{10} = 50$  ms, with  $\psi = 0.167$  and  $\gamma = 3.76$  so that the peaks were evenly spaced in logarithmic time according to  $\gamma \log \left( \frac{\phi_{j+1}+\psi}{\phi_j+\psi} \right) = \frac{\pi}{2}$ . This basis allows for fine temporal structure on short timescales and coarse temporal structure on longer timescales, and precludes temporal aliasing (see (Pillow et al., 2005; Pillow et al., 2008)). The constant  $b$  was 3.1 for the OFF cell and 2.25 for the ON cell, corresponding to baseline spike rates of 20Hz and 9.5Hz, respectively.

For analyses involving populations of simulated neurons (figs. 2, 3, 4, 7, 8 10, and

12), identical copies of these two neurons were created. For the spatial-decoding analysis shown in Fig. (7), the neurons were equipped with canonical difference-of-Gaussian spatial receptive fields (with standard deviations of center and surround given by  $\sigma_{ctr} = 0.75$  pixels and  $\sigma_{sur} = 1.5$  pixels, respectively, with weighting 1 and  $-.25$  for center and surround). For the larger image example (Fig. 7B, these spatial receptive fields were scaled up by a factor of 4 in order to achieve identical tiling of the image plane. For figures 9 and 11, a slightly different set of parameters was used, with symmetric, larger-amplitude stimulus filters and identical post-spike filters that induced strong refractory effects (shown in Fig. 13B).

For all examples, spike trains from the GLM were sampled using the time-rescaling transform (Brown et al., 2002), with time discretized in bins of size 0.08 ms. MAP decoding was carried out assuming knowledge the true GLM parameters. (The results do not differ if these parameters are instead estimated from a sufficiently large set of simulated responses). The optimal linear filter (OLE, eq. 16) was estimated by using simulated GLM responses to a 300,000-sample (42-minute) stimulus to estimate the terms  $\mathbb{E}[\mathbf{r}\mathbf{r}^T]$  and  $\mathbb{E}[\mathbf{r}^T x]$ . The OLE filter was estimated separately for each contrast level and number of neurons.

Except where otherwise noted, the stimulus prior  $p(\mathbf{x})$  used for MAP decoding was standard (independent, zero-mean, unit-variance) Gaussian. For the one-dimensional example showing performance under a naturalistic (“1/F”) prior (Fig. 4), the prior was taken to be Gaussian with unit marginal variance and a covariance that was diagonal in the Fourier domain, with standard deviation proportional to 1 over frequency (leading to a highly correlated prior in the time domain). For the spatial decoding of natural images (Fig. 7), the prior was diagonal in the 2D Fourier domain, with marginal variance set to the true variance of the pixel intensities (which affects only the total scaling of the MAP estimate) and standard deviation falling as  $1/F^2$ .

## Acknowledgments

Thanks to M. Berry, D. Butts, E.J. Chichilnisky, P. Dayan, P. Latham, H. Liu, S. Nirenberg, A. Ramirez, E. Simoncelli and S. Tripathy for helpful comments and conversations. JP is funded by an International Research Fellowship from the Royal Society and the Center for Perceptual Systems at the University of Texas at Austin; YA is supported by a Robert Leet and Clara Guthrie Patterson Trust Postdoctoral Fellowship, Bank of America, Trustee; LP is supported by NEI grant EY018003 and by a McKnight Scholar award.

## References

- Abbott, L. and Dayan, P. (1999). The effect of correlated variability on the accuracy of a population code. *Neural Computation*, 11:91–101.
- Ahmadian, Y., Pillow, J., and Paninski, L. (2010). Efficient Markov Chain Monte Carlo methods for decoding population spike trains. *To Appear, Neural Computation*.
- Barbieri, R., Frank, L., Nguyen, D., Quirk, M., Solo, V., Wilson, M., and Brown, E. (2004). Dynamic analyses of information encoding in neural ensembles. *Neural Computation*, 16:277–307.
- Berry, M. and Meister, M. (1998). Refractoriness and neural precision. *Journal of Neuroscience*, 18:2200–2211.
- Bialek, W., Rieke, F., de Ruyter van Steveninck, R., R. R., and Warland, D. (1991). Reading a neural code. *Science*, 252:1854–1857.
- Bialek, W. and Zee, A. (1990). Coding and computation with neural spike trains. *Journal of Statistical Physics*, 59:103–115.

- Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Oxford University Press.
- Brillinger, D. (1988). Maximum likelihood analysis of spike trains of interacting nerve cells. *Biological Cyberkinetics*, 59:189–200.
- Brillinger, D. (1992). Nerve cell spike train data analysis: a progression of technique. *Journal of the American Statistical Association*, 87:260–271.
- Brockwell, A., Rojas, A., and Kass, R. (2004). Recursive Bayesian decoding of motor cortical signals by particle filtering. *Journal of Neurophysiology*, 91:1899–1907.
- Brockwell, A. E., Kass, R. E., and Schwartz, A. (2007). Statistical signal processing and the motor cortex. *Proceedings of the IEEE*, 95:1–18.
- Brown, E., Barbieri, R., Ventura, V., Kass, R., and Frank, L. (2002). The time-rescaling theorem and its application to neural spike train data analysis. *Neural Computation*, 14:325–346.
- Brown, E., Frank, L., Tang, D., Quirk, M., and Wilson, M. (1998). A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells. *Journal of Neuroscience*, 18:7411–7425.
- Brunel, N. and Nadal, J. P. (1998). Mutual information, fisher information, and population coding. *Neural Computation*, 10:773–82.
- Butts, D. and Stanley, G. (2003). Quick and accurate information calculations based on linear characterizations of sensory neurons. *Society for Neuroscience Abstracts*.
- Chacron, M. (2005). Nonlinear information processing in a model sensory system. *Journal of Neurophysiology*, 95:2933–2946.

- Chichilnisky, E. J. (2001). A simple white noise analysis of neuronal light responses. *Network: Computation in Neural Systems*, 12:199–213.
- Clarke, B. and Barron, A. (1990). Information-theoretic asymptotics of Bayes methods. *IEEE Transactions on Information Theory*, 36:453 – 471.
- Cover, T. and Thomas, J. (1991). *Elements of information theory*. Wiley, New York.
- Cronin, B., Stevenson, I. H., Sur, M., and Kording, K. P. (2009). Hierarchical bayesian modeling and markov chain monte carlo sampling for tuning curve analysis. *J Neurophysiol*, page 00379.2009.
- Dayan, P. and Abbott, L. (2001). *Theoretical Neuroscience*. MIT Press.
- de Ruyter, R. and Bialek, W. (1988). Real-time performance of a movement-sensitive neuron in the blowfly visual system: coding and information transmission in short spike sequences. *Proc. R. Soc. Lond. B*, 234:379–414.
- de Ruyter van Steveninck, R. and Bialek, W. (1995). Reliability and statistical efficiency of a blowfly movement-sensitive neuron. *Phil Trans R Soc Lond Ser B*, 348:321–340.
- DeWeese, M. and Zador, A. (1998). Asymmetric dynamics in optimal variance adaptation. *Neural Computation*, 10(5):1179–1202.
- Donoghue, J. (2002). Connecting cortex to machines: recent advances in brain interfaces. *Nature Neuroscience*, 5:1085–1088.
- Doucet, A., de Freitas, N., and Gordon, N., editors (2001). *Sequential Monte Carlo in Practice*. Springer.
- Duda, R. and Hart, P. (1972). *Pattern classification and scene analysis*. Wiley, New York.

- Eichhorn, J., Tolias, A., Zien, A., Kuss, M., Rasmussen, C., Weston, J., Logothetis, N., and Schoelkopf, B. (2004). Prediction on spike data using kernel algorithms. *NIPS*, 16.
- Ergun, A., Barbieri, R., Eden, U., Wilson, M., and Brown, E. (2007). Construction of point process adaptive filter algorithms for neural systems using sequential Monte Carlo methods. *IEEE Transactions on Biomedical Engineering*, 54:419–428.
- Fahrmeir, L. (1992). Posterior mode estimation by extended Kalman filtering for multivariate dynamic generalized linear models. *Journal of the American Statistical Association*, 87(418):501–509.
- Gerwinn, S., Macke, J., and Bethge, M. (2009). Bayesian population decoding of spiking neurons. *Frontiers in Computational Neuroscience*, 3:1–28.
- Gerwinn, S., Macke, J., Seeger, M., and Bethge, M. (2008). Bayesian inference for spiking neuron models with a sparsity prior. *Advances in Neural Information Processing Systems*.
- Haag, J. and Borst, A. (1997). Encoding of visual motion information and reliability in spiking and graded potential neurons. *Journal of Neuroscience*, 17:4809–4819.
- Harris, K., Csicsvari, J., Hirase, H., Dragoi, G., and Buzsaki, G. (2003). Organization of cell assemblies in the hippocampus. *Nature*, 424:552–556.
- Humphrey, D., Schmidt, E., and Thompson, W. (1970). Predicting measures of motor performance from multiple cortical spike trains. *Science*, 170:758–762.
- Huys, Q. J. M., Ahrens, M. B., and Paninski, L. (2006). Efficient estimation of detailed single-neuron models. *J Neurophysiol*, 96(2):872–890.



- Jacobs, A., Grzywacz, N., and Nirenberg, S. (2006). Decoding the parallel pathways of the retina. *SFN Abstracts*.
- Jordan, M. I., editor (1999). *Learning in graphical models*. MIT Press, Cambridge, MA, USA.
- Karmeier, K., Krapp, H., and Egelhaaf, M. (2005). Population coding of self-motion: Applying bayesian analysis to a population of visual interneurons in the fly. *Journal of Neurophysiology*, 94:2182–2194.
- Kass, R. and Raftery, A. (1995). Bayes factors. *Journal of the American Statistical Association*, 90:773–795.
- Kelly, R. and Lee, T. (2004). Decoding V1 neuronal activity using particle filtering with Volterra kernels. *Advances in Neural Information Processing Systems*, 15:1359–1366.
- Kennel, M., Shlens, J., Abarbanel, H., and Chichilnisky, E. (2005). Estimating entropy rates with Bayesian confidence intervals. *Neural Computation*, 17:1531–1576.
- Koyama, S. and Paninski, L. (2009). Efficient computation of the maximum a posteriori path and parameter estimation in integrate-and-fire and more general state-space models. *Journal of Computational Neuroscience*, In press.
- Kulkarni, J. E. and Paninski, L. (2007). Common-input models for multiple neural spike-train data. *Network: Computation in Neural Systems*, 18(4):375–407.
- Latham, P. and Nirenberg, S. (2005). Synergy, redundancy, and independence in population codes, revisited. *J. Neurosci.*, 25:5195–5206.
- Lazar, A. A. and Pnevmatikakis, E. A. (2008). Faithful representation of stimuli with a population of integrate-and-fire neurons. *Neural Computation*, 20(11):2715–2744.

- Lyu, S. and Simoncelli, E. P. (2009). Nonlinear extraction of 'independent components' of natural images using radial Gaussianization. *Neural Computation*, 21(6):1485–1519.
- Marmarelis, P. Z. and Marmarelis, V. (1978). *Analysis of physiological systems: the white-noise approach*. Plenum Press, New York.
- Maynard, E., Hatsopoulos, N., Ojakangas, C., Acuna, B., Sanes, J., Normann, R., and Donoghue, J. (1999). Neuronal interactions improve cortical population coding of movement direction. *Journal of Neuroscience*, 19:8083–8093.
- McCullagh, P. and Nelder, J. (1989). *Generalized linear models*. Chapman and Hall, London.
- Mesgarani, N., David, S. V., Fritz, J. B., and Shamma, S. A. (2009). Influence of context and behavior on stimulus reconstruction from neural activity in primary auditory cortex. *J Neurophysiol*, 102(6):3329–3339.
- Minka, T. (2001). *A Family of Algorithms for Approximate Bayesian Inference*. PhD thesis, MIT.
- Nemenman, I., Bialek, W., and de Ruyter van Steveninck, R. (2004). Entropy and information in neural spike trains: Progress on the sampling problem. *Physical Review E*, 69:056111.
- Nicolelis, M., Dimitrov, D., Carmena, J., Crist, R., Lehew, G., Kralik, J., and Wise, S. (2003). Chronic, multisite, multielectrode recordings in macaque monkeys. *PNAS*, 100:11041–11046.
- Paninski, L. (2003). Estimation of entropy and mutual information. *Neural Computation*, 15:1191–1253.

- Paninski, L. (2004). Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems*, 15:243–262.
- Paninski, L. (2005). Asymptotic theory of information-theoretic experimental design. *Neural Computation*, 17(7):1480–1507.
- Paninski, L. (2006). The most likely voltage path and large deviations approximations for integrate-and-fire neurons. *Journal of Computational Neuroscience*, 21:71–87.
- Paninski, L., Ahmadian, Y., Ferreira, D., Koyama, S., Rahnama Rad, K., Vidne, M., Vogelstein, J., and Wu, W. (2010). A new look at state-space models for neural data. *Journal of Computational Neuroscience*, In press.
- Paninski, L., Fellows, M., Shoham, S., Hatsopoulos, N., and Donoghue, J. (2004a). Super-linear population encoding of dynamic hand trajectory in primary motor cortex. *J. Neurosci.*, 24:8551–8561.
- Paninski, L., Pillow, J. W., and Lewi, J. (2007). Statistical models for neural encoding, decoding, and optimal stimulus design. In Cisek, P., Drew, T., and Kalaska, J., editors, *Computational Neuroscience: Theoretical Insights Into Brain Function*, Progress in Brain Research, pages 493–507. Elsevier.
- Paninski, L., Pillow, J. W., and Simoncelli, E. P. (2004b). Maximum likelihood estimation of a stochastic integrate-and-fire neural model. *Neural Computation*, 16:2533–2561.
- Paninski, L., Pillow, J. W., and Simoncelli, E. P. (2005). Comparing integrate-and-fire-like models estimated using intracellular and extracellular data. *Neurocomputing*, 65–66:379–385.
- Passaglia, C. and Troy, J. (2004). Information transmission rates of cat retinal ganglion cells. *Journal of Neurophysiology*, 91:1217–1229.

- Pillow, J. (2009). Time-rescaling methods for the estimation and assessment of non-poisson neural encoding models. In Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C. K. I., and Culotta, A., editors, *Advances in Neural Information Processing Systems 22*, pages 1473–1481. MIT Press.
- Pillow, J. W., Paninski, L., and Simoncelli, E. P. (2004). Maximum likelihood estimation of a stochastic integrate-and-fire neural model. In S. Thrun, L. S. and Scholkopf, B., editors, *Advances in Neural Information Processing Systems*, volume 16.
- Pillow, J. W., Paninski, L., Uzzell, V. J., Simoncelli, E. P., and Chichilnisky, E. J. (2005). Prediction and decoding of retinal ganglion cell responses with a probabilistic spiking model. *The Journal of Neuroscience*, 25:11003–11013.
- Pillow, J. W., Shlens, J., Paninski, L., Sher, A., Litke, A. M., and Chichilnisky, E. J. Simoncelli, E. P. (2008). Spatio-temporal correlations and visual signaling in a complete neuronal population. *Nature*, 454:995–999.
- Pillow, J. W. and Simoncelli, E. P. (2003). Biases in white noise analysis due to non-Poisson spike generation. *Neurocomputing*, 52:109–115.
- Pillow, J. W. and Simoncelli, E. P. (2006). Dimensionality reduction in neural models: An information-theoretic generalization of spike-triggered average and covariance analysis. *Journal of Vision*, 6(4):414–428.
- Plesser, H. and Gerstner, W. (2000). Noise in integrate-and-fire neurons: From stochastic input to escape rates. *Neural Computation*, 12:367–384.
- Press, W., Teukolsky, S., Vetterling, W., and Flannery, B. (1992). *Numerical recipes in C*. Cambridge University Press.

- Rieke, F., Warland, D., de Ruyter van Steveninck, R. R., and Bialek, W. (1997). *Spikes: Exploring the Neural Code*. MIT Press, Cambridge, MA.
- Rigat, F., de Gunst, M., and van Pelt, J. (2006). Bayesian modelling and analysis of spatio-temporal neuronal networks. *Bayesian Analysis*, 1(4):733–764.
- Robert, C. and Casella, G. (2005). *Monte Carlo Statistical Methods*. Springer.
- Sahani, M. (2000). Kernel regression for neural systems identification. Presented at NIPS00 workshop on Information and statistical structure in spike trains; abstract available at <http://www-users.med.cornell.edu/~jdvicto/nips2000speakers.html>.
- Salinas, E. and Abbott, L. (2001). *Principles of Neural Ensemble and Distributed Coding in the Nervous System*, chapter Coordinate transformations in the visual system: how to generate gain fields and what to compute with them, pages 175–190. Elsevier.
- Sanger, T. (1994). Theoretical considerations for the analysis of population coding in motor cortex. *Neural Computation*, 6:12–21.
- Schervish, M. (1995). *Theory of statistics*. Springer-Verlag, New York.
- Scholkopf, B. and Smola, A. (2002). *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press.
- Serruya, M., Hatsopoulos, N., Paninski, L., Fellows, M., and Donoghue, J. (2002). Instant neural control of a movement signal. *Nature*, 416:141–142.
- Sharpee, T., Rust, N., and Bialek, W. (2004). Analyzing neural responses to natural signals: Maximally informative dimensions. *Neural Computation*, 16:223–250.
- Shoham, S., Paninski, L., Fellows, M., Hatsopoulos, N., Donoghue, J., and Normann, R. (2005). Optimal decoding for a primary motor cortical brain-computer interface. *IEEE Transactions on Biomedical Engineering*, 52:1312–1322.

- Shpigelman, L., Singer, Y., Paz, R., and Vaadia, E. (2003). Spikernels: embedding spike neurons in inner product spaces. *NIPS*, 15.
- Simoncelli, E. P. (2005). Statistical modeling of photographic images. In Bovik, A., editor, *Handbook of Image and Video Processing*, chapter 4.7, pages 431–441. Academic Press. 2nd edition.
- Simoncelli, E. P., Paninski, L., Pillow, J., and Schwartz, O. (2004). Characterization of neural responses with stochastic stimuli. In Gazzaniga, M., editor, *The Cognitive Neurosciences*, pages 327–338. MIT Press, 3rd edition.
- Srinivasan, L., Eden, U., Willsky, A., and Brown, E. (2006). A state-space analysis for reconstruction of goal-directed movements using neural signals. *Neural Computation*, 18:2465–2494.
- Stanley, G. and Bolori, A. (2001). Decoding in neural systems: stimulus reconstruction from nonlinear encoding. *Proceedings of the Annual EMBS International Conference*, 23.
- Stevenson, I., Rebesco, J., Hatsopoulos, N., Haga, Z., Miller, L., and Kording, K. (2009). Bayesian inference of functional connectivity and network structure from spikes. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 17(3):203.
- Strong, S., Koberle, R., de Ruyter van Steveninck R., and Bialek, W. (1998). Entropy and information in neural spike trains. *Physical Review Letters*, 80:197–202.
- Theunissen, F., Roddey, J., Stufflebeam, S., Clague, H., and Miller, J. (1996). Information theoretic analysis of dynamical encoding by four primary sensory interneurons in the cricket cercal system. *Journal of Neurophysiology*, 75:1345–1364.

- Truccolo, W., Eden, U. T., Fellows, M. R., Donoghue, J. P., and Brown, E. N. (2005). A point process framework for relating neural spiking activity to spiking history, neural ensemble and extrinsic covariate effects. *J. Neurophysiol*, 93(2):1074–1089.
- Truccolo, W., Hochberg, L., and Donoghue, J. (2010). Collective dynamics in human and monkey sensorimotor cortex: predicting single neuron spikes. *Nature neuroscience*, 13:105–111.
- Warland, D., Reinagel, P., and Meister, M. (1997). Decoding visual information from a population of retinal ganglion cells. *Journal of Neurophysiology*, 78:2336–2350.
- Wu, W., Black, M. J., Mumford, D., Gao, Y., Bienenstock, E., and Donoghue, J. P. (2004). Modeling and decoding motor cortical activity using a switching Kalman filter,. *IEEE Transactions on Biomedical Engineering*, 51:933–942.
- Wu, W., Kulkarni, J., Hatsopoulos, N., and Paninski, L. (2009). Neural decoding of goal-directed movements using a linear statespace model with hidden states. *IEEE Trans. Biomed. Eng.*, In press.
- Yu, B. M., Cunningham, J. P., Shenoy, K. V., and Sahani, M. (2007). Neural decoding of movements: From linear to nonlinear trajectory models. *ICONIP*, pages 586–595.
- Zhang, K., Ginzburg, I., McNaughton, B., and Sejnowski, T. (1998). Interpreting neuronal population activity by reconstruction: Unified framework with application to hippocampal place cells. *Journal of Neurophysiology*, 79:1017–1044.

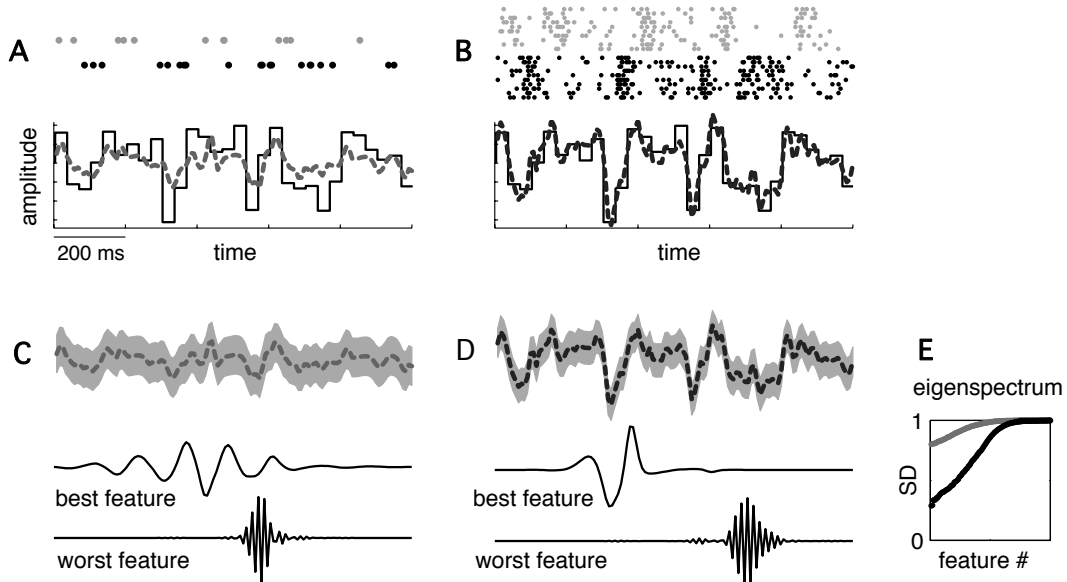


Figure 2: Illustration of MAP decoding on a 1-second Gaussian white-noise stimulus. **(A)** Spike times generated by a single ON cell (gray dots) and a single OFF cell (black dots) in simulation. Black solid trace shows the true stimulus and gray dashed trace is the *maximum a posteriori* (MAP) estimate  $\mathbf{x}_{\text{map}}$  given these spikes (estimated on a time lattice four times finer than the stimulus lattice). **(B)** Simulated spike trains from 10 ON and 10 OFF neurons in response to the same stimulus, and the corresponding MAP estimate (dashed). **(C-D)** MAP estimates under 2-neuron and 20-neuron populations, replotted with gray region representing  $\pm 1$  standard deviation of the posterior uncertainty about stimulus value. (These error-bars may also be computed in  $O(T)$  time (Paninski et al., 2010).) Below: stimulus features that are best- and worst-constrained by the observed spike data, determined as the eigenvectors of the inverse Hessian with smallest and largest eigenvalue, respectively. Perturbing  $\mathbf{x}_{\text{map}}$  with the best (worst) feature causes the fastest (slowest) falloff in the posterior. **(E)** Eigenspectrum of  $J^{-1}$  at the MAP estimate from both populations (gray=2; black = 20 neurons). The sorted (square-roots of the) eigenvalues characterize the standard deviation of the posterior  $p(\mathbf{x}|\mathbf{r})$  along stimulus axes defined by the corresponding eigenvectors. Note that the high eigenvalues (corresponding to high-frequency stimulus information) remain poorly constrained even in the 20-neuron case, due to the lowpass nature of the stimulus filter  $\mathbf{k}$ .



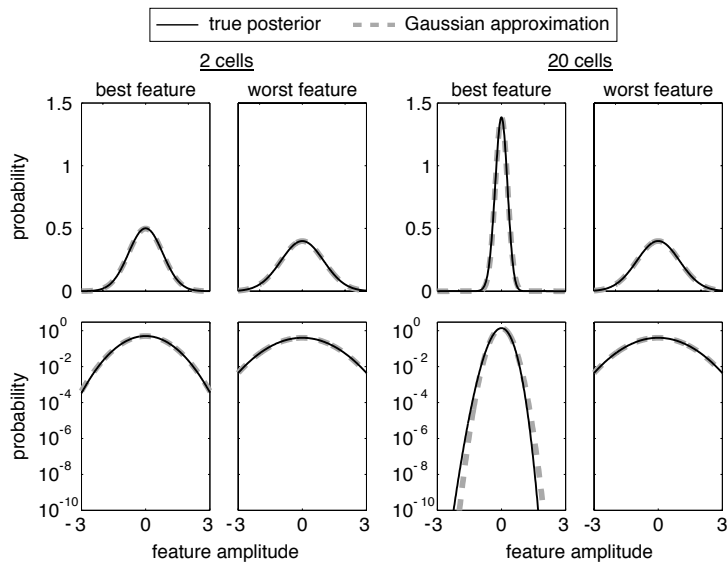


Figure 3: Comparison of true posterior (solid) with Gaussian approximation (dashed). Top Row: 1D slices through  $p(\mathbf{x}|\mathbf{r})$  and the Gaussian approximation along axes corresponding to best and most poorly encoded stimulus features (i.e., axes of least and greatest posterior variance), for the 2-cell and 20-cell responses shown in Fig. 2A (left) and 2B (right). Bottom row: same distributions plotted on a log-scale, showing some mismatch in the tails and skewness as the posterior grows sharper.

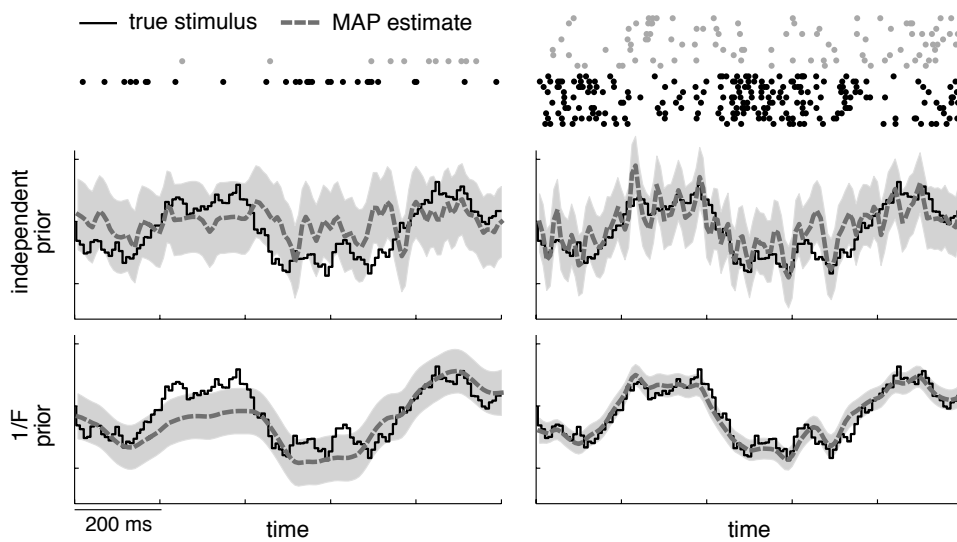


Figure 4: Illustration of MAP decoding under a correlated prior. Top left: spike response of an ON cell (gray dots) and OFF cell (black dots) to a Gaussian stimulus with  $1/F$  temporal correlation structure. Middle: True stimulus (black) and MAP estimate (dashed) under an independent prior. Bottom: True stimulus and MAP estimate under the correct ( $1/F$ ) stimulus prior. Right: MAP estimates computed from the responses of 20-neuron population. Light gray regions show  $\pm 1\text{SD}$  confidence interval, computed using the square root of the diagonal elements of the inverse Hessian matrix.

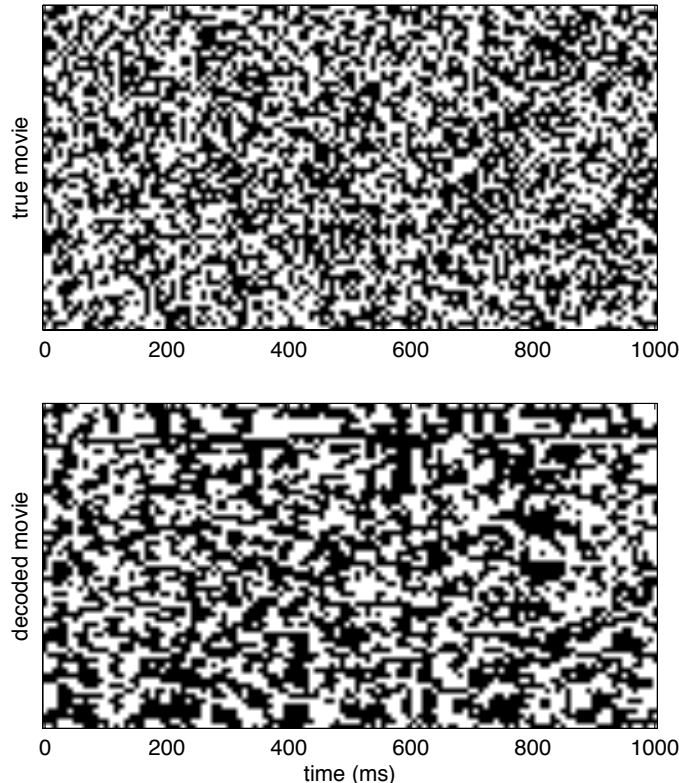


Figure 5: Decoding of a binary white noise movie based on experimentally recorded spike trains of a group of 27 retinal ganglion cells. The stimulus was an 8 pixel by 8 pixel movie, in which at every time step (the movie refresh rate was 120 Hz) the intensity of each pixel was independently sampled from a binary distribution with a contrast of 0.5. We decoded a portion of the movie with a duration of 1 second ( $\dim(\mathbf{x})$  here was therefore  $120 \times 64 = 7680$ ), based on the recorded spike trains of 16 OFF and 11 ON cells whose receptive fields imperfectly covered the  $8 \times 8$  pixel movie area. The GLM parameters used in the decoding were previously fit to 20 minutes of the same data (see (Pillow et al., 2008) for the details of the experimental recordings and the fit model parameters). The top and bottom panels show the true and decoded movies, respectively. Each vertical slice shows the contrast of the 64 pixels, linearly ordered, at the corresponding time step. The positive (negative) contrast pixels are shown in white (black). To find the MAP using the fast  $O(T)$  methods, we used a log-concave flat prior with support on  $[-c, c]$  (where  $c = 0.5$  is the contrast) along every dimension. To obtain an estimate with binary values at each pixel, we then truncated each component of the thus obtained MAP to the nearer value in  $\{-c, c\}$ . Since the cell receptive field centers did not cover the entire movie area, and the movie had a relatively fast refresh rate (well above the effective temporal bandlimit imposed by the stimulus filter matrix  $K$ ), the signal to noise ratio was relatively low, and therefore the decoding error was high. The Hamming distance, per frame per pixel, between the true and the decoded movies was 0.43. (See (Ahmadian et al., 2010; Paninski et al., 2010) for further examples, including a demonstration that the filtered stimulus  $K\mathbf{x}$  can in fact be decoded quite accurately here.)

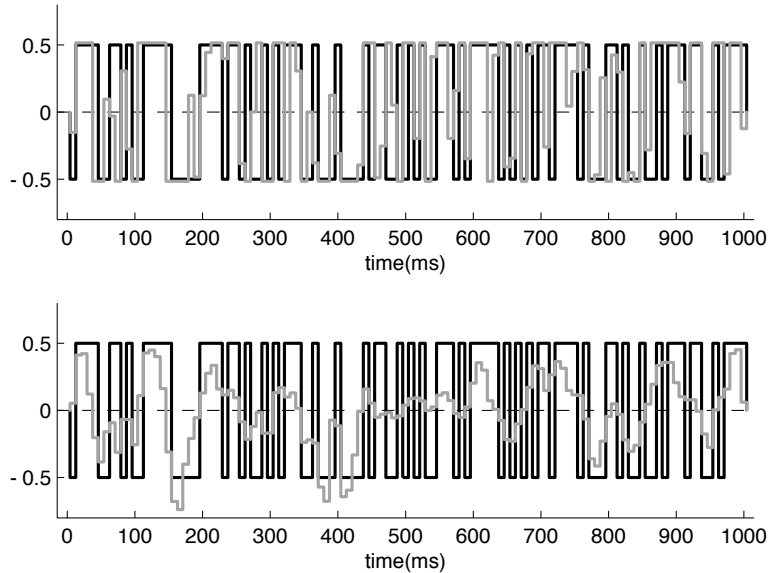


Figure 6: A pixel from the movie decoding example of Fig. 5 above, singled out to illustrate the effect of the prior on binary stimulus decoding. Plots show the un-truncated decoders' estimates (gray traces) for the true time-varying contrast (black traces) of the selected pixel. The MAP estimate was obtained using a flat prior (**top**) and a Gaussian prior with the same contrast (std = 0.5, **bottom**). In regions where information carried by the spike trains regarding the true stimulus is low (e.g., due to low firing rate), the decoder with Gaussian prior shrinks the estimate towards the prior mean (zero), whereas the flat prior decoder tends to stick to the boundaries of the prior support. (For better visual clarity, in the top panel, we have shifted the flat prior decoder's estimate up by a small factor, so that it does not cover the true stimulus curve when it does in fact coincide with it.) For this pixel, MAP decoding achieves an SNR of 0.82 under a flat prior, vs. an SNR of 1.25 under a Gaussian prior, indicating that shrinkage induced by the Gaussian prior is effective in reducing the mean-squared error of the MAP estimate.

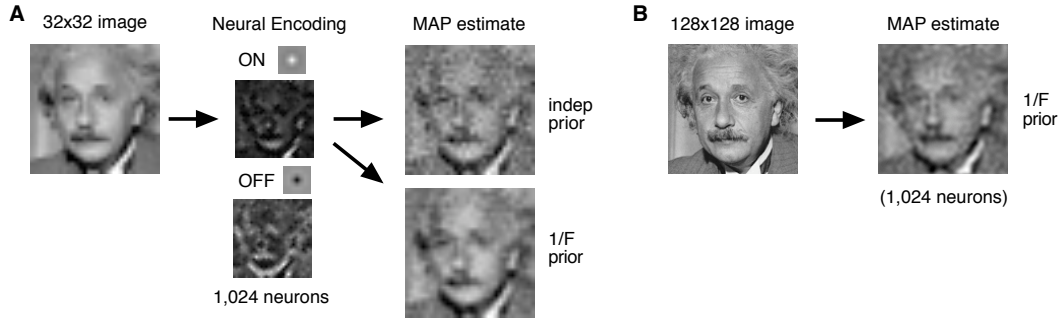


Figure 7: Decoding of a high-dimensional naturalistic spatial image. **(A)** Decoding of a  $32 \times 32$  image from the responses of a population of 1,024 simulated retinal ganglion cells. Intensity plot (center) shows the spike count of each neuron (512 ON, 512 OFF cells with center-surround receptive fields spanning the 1,024-dimensional stimulus space) in response to a brief (0.5s) presentation of the image. Right: estimate of the stimulus based on MAP decoding of the population response, under an independent Gaussian prior (above) and Gaussian prior with  $1/F^2$  spatial correlations (below). **(B)** MAP decoding in a stimulus space with higher dimensionality ( $\dim(\mathbf{x}) = 16,384$ ) than the number of neurons (1,024). Receptive fields were spatially up-sampled by a factor of four so that tiling of the image plane matched that in (A). Decoding under a  $1/F^2$  Gaussian prior over the full stimulus space (right) leads to higher-fidelity decoding than for the  $32 \times 32$  image, despite having the same low-dimensional representation of the signal.

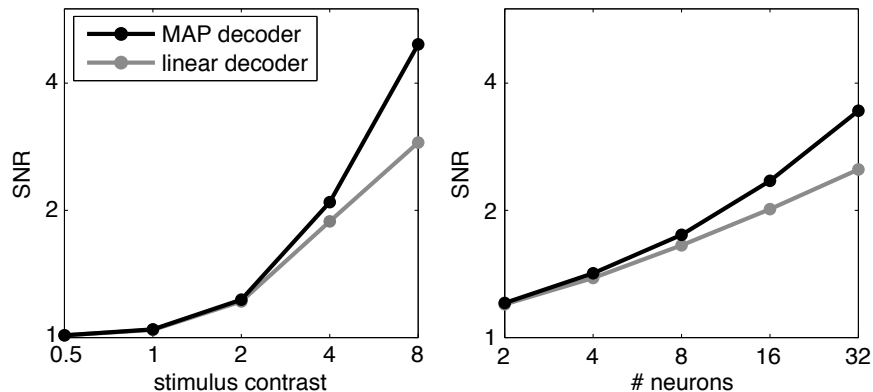


Figure 8: Comparison of MAP and optimal linear decoding of simulated retinal ganglion cell responses. Stimuli were 0.5s segments of Gaussian white noise sampled at a frame rate of 15 Hz. The optimal linear decoding filter was fit via linear regression, for each contrast and population size. Performance was measured as the signal-noise-ratio, calculated as the ratio of the signal variance to the variance of the reconstruction error, averaged over 200 stimulus segments drawn i.i.d. from the prior. **Left:** Decoding performance as a function of stimulus standard deviation, using two cells (an uncoupled ON and OFF neuron). **Right:** Decoding performance as a function of population size, using a stimulus with contrast  $c = 2$ . Each population contained an equal number of ON and OFF cells.

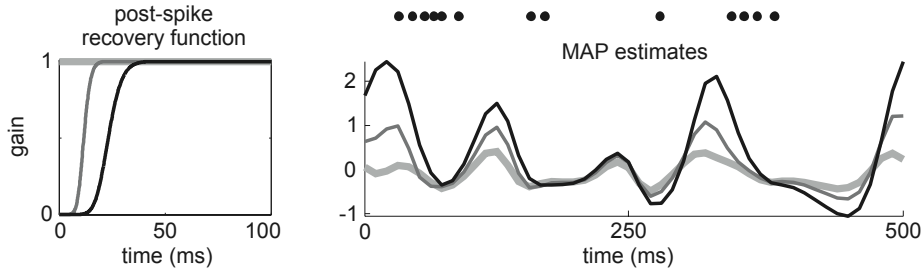


Figure 9: **Left:** Three different post-spike recovery functions (exponentiated post-spike kernels), which multiply the conditional intensity function following a spike. These induce spike-history effects ranging from “none” (light gray) to a relative refractory period of approximately 25ms (black). **Right:** MAP decoding of a single set of spike times (dots) under three GLMs that differ only in their post-spike kernels (shown at left). Spike bursts are interpreted quite differently by the three models, indicating large stimulus transients under the model with strong refractory effects (since for a burst to have occurred the stimulus must have been large enough to overcome the refractory effects), whereas isolated spikes (i.e., near 250ms) have nearly the same decoded interpretation for all three models.

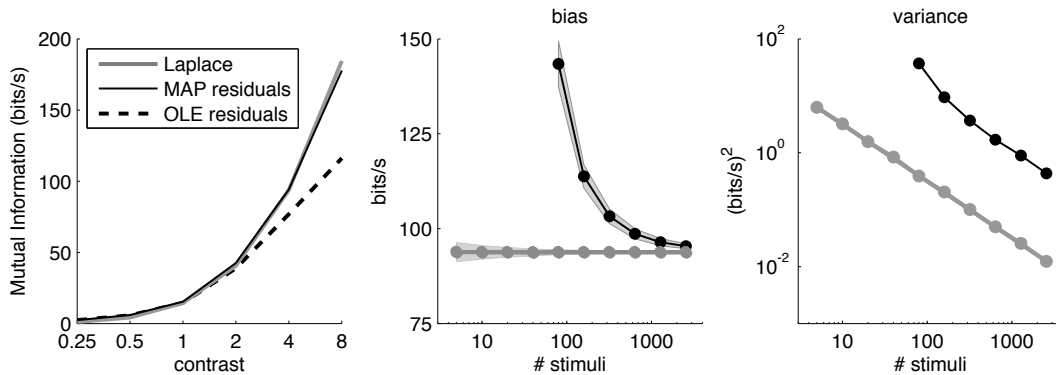


Figure 10: Model-based estimates of mutual information (MI). **Left:** Estimates of the information rate between a 60-dimensional stimulus (Gaussian white noise) and the spike responses from a 32-neuron population (16 ON and 16 OFF simulated GLM neurons, with parameters fit to the data described in (Pillow et al., 2008)), as a function of stimulus standard deviation (“contrast”). The OLE and MAP residuals (black solid and dashed traces) can be used to compute a lower bound on the MI (see text). The Laplace approximation can be used to estimate MI by the averaging the posterior entropy across stimulus-response pairs (gray trace). **Middle:** Bias in the MAP residual-based (black) and Laplace-based (gray) estimators of MI at contrast=4, as a function of the number of stimulus-response pairs. Note that the residual-based estimate (black) does not actually provide a lower bound on MI unless the estimate of the residual covariance matrix has converged. (When  $[\# \text{ stimuli}] < [\# \text{ dimensions}]$ , the residual-based estimate is actually infinite). Grey regions show 4 standard deviations of the noise in estimating MI under both methods, based on 2000 bootstrap resamplings of the data. **Right:** Variance of the same two estimators as a function of the amount of data used.

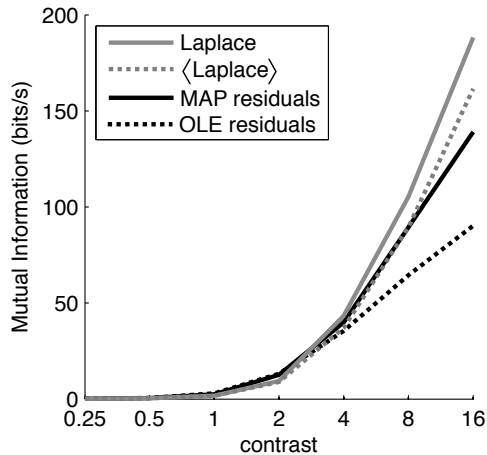


Figure 11: Mutual information (MI) estimates for a population of two highly-nonlinear neurons. Responses to Gaussian white noise stimuli were simulated from an ON/OFF pair of neurons with larger-amplitude stimulus filters and longer refractory periods than neurons used in Fig. 10. The MI between 0.25s stimulus segments and the spike response was estimated using the OLE residuals, MAP residuals and the posterior entropy under the Laplace approximation. At the highest contrast, the Laplace approximation-based estimate (gray trace) is 35% higher than the lower bound defined by the MAP residuals (black). Looseness of the lower bound is at least partly explained by trial-to-trial fluctuations in the posterior, since the estimate formed by averaging the Hessian across all stimulus-response pairs (labelled “ $\langle$ Laplace $\rangle$ ”, dotted gray trace) is only 16% above the MAP-based lower bound. The  $\langle$ Laplace $\rangle$  estimate is equivalent to assuming that the posterior is Gaussian with a single fixed covariance (estimated as the average of the posterior covariances on each trial; see text). This shows that the inequality (eq. 22) is not tight in certain cases.

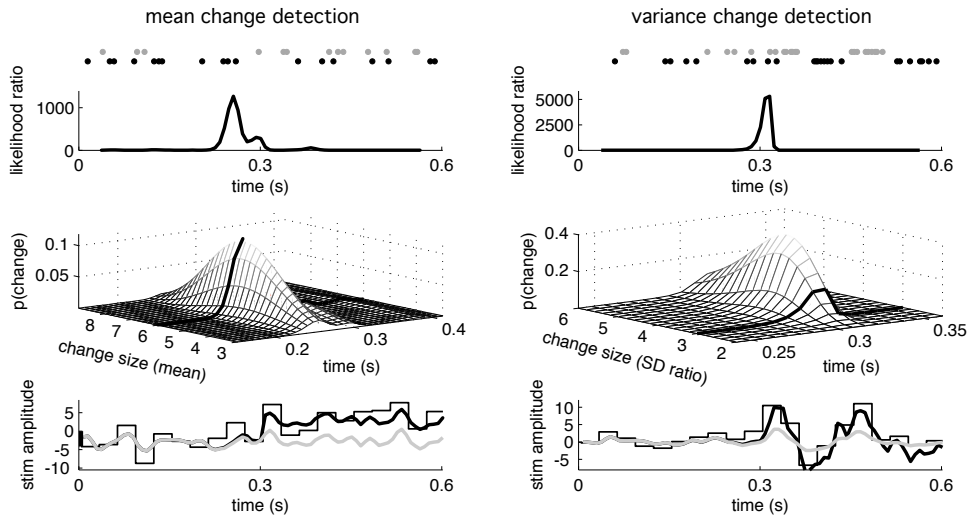


Figure 12: Illustration of change-point detection for changes in mean (left; change from  $\mu = -3$  to  $\mu = 3$ , with fixed stdev  $\sigma = 3$ ) and variance (right; change from  $\sigma = 2$  to  $\sigma = 6$ ). **(Top:)** Spike times emitted by a simulated pair of (ON and OFF) retinal ganglion cells in response to a 0.6s stimulus whose mean / variance undergoes a change at time  $t = 0.3$ s. The organism’s task is to determine, from these spike times, if and when a change occurred, and if so how large it was. **Second row:** Log-likelihood ratio of the hypothesis “Change in mean at time  $t$ ” (eq. 29) to that of “no change”, plotted as a function of  $t$ , assuming that we know the expected change size. **Third row:** A two-dimensional posterior distribution over time and change size if we do not assume we know the latter. Black line shows posterior probability of change for the true step size (identical to black trace on second row). Note that the change-time and step size are inferred fairly accurately, though the variance change would be overestimated. **Bottom row** True stimulus (thin black) and MAP estimate using the correct change time and height (thick black) or assuming no change (gray).



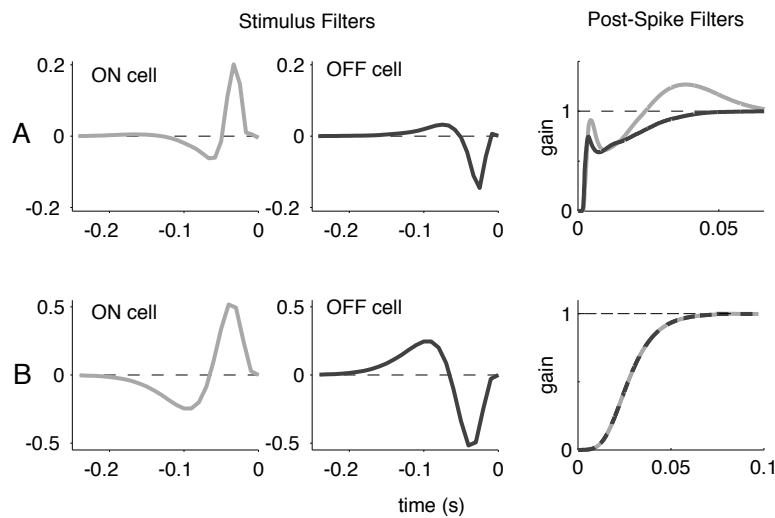


Figure 13: GLM parameters used for simulating spike trains and MAP-based decoding and applications. **(A)** Parameters fit to one ON and one OFF retinal ganglion cell (for methods, see (Pillow et al., 2008)). Filter parameters (left) describe the integration of light as a function of time before a spike. Exponentiated post-spike filters (right) show the multiplicative effect of a spike at time zero on the subsequent probability of spiking in either cell. **(B)** Parameters used for analyses shown in figs. 9 and 11. Larger-amplitude filters and longer relative refractory period make responses more nonlinear.