J Comput Neurosci
DOI 10.1007/s10827-009-0157-3

Feature extraction from spike trains with Bayesian binning: 'Latency is where the signal starts'

Dominik Endres · Mike Oram

Received: 13 November 2008 / Revised: 2 March 2009 / Accepted: 14 April 2009

Abstract The peristimulus time histogram (PSTH) and its more continuous cousin, the spike density function (SDF) are staples in the analytic toolkit of neurophysiologists. The former is usually obtained by binning spike trains, whereas the standard method for the latter is smoothing with a Gaussian kernel. Selection of a bin width or a kernel size is often done in an relatively arbitrary fashion, even though there have been recent attempts to remedy this situation (DiMatteo et al., Biometrika 88(4):1055–1071, 2001; Shimazaki and Shinomoto 2007a, Neural Comput 19(6):1503–1527, 2007b, c; Cunningham et al. 2008). We develop an exact Bayesian, generative model approach to estimating PSTHs. Advantages of our scheme include automatic complexity control and error bars on its predictions. We show how to perform feature extraction on spike trains in a principled way, exemplified through latency and firing rate posterior distribution evaluations on repeated and single trial data. We also demonstrate using both simulated and real neuronal data that our approach provides a more accurate estimates of the PSTH and the latency than current competing methods. We employ the posterior distributions for an information theoretic analysis of the neural code comprised of latency and firing rate of neurons in high-level visual area STSa. A software implementation of our method is available at the machine learning open source software repository (www.mloss.org, project 'binsdfc').

Keywords Spike train analysis · Bayesian methods · Response latency · PSTH · SDF · Information theory

Action Editor: Matthew Wiener

D. Endres (✉) · M. Oram
School of Psychology, University of St. Andrews,
St. Andrews, KY16 9JP, UK
e-mail: dme2@st-andrews.ac.uk

M. Oram
e-mail: mwo@st-andrews.ac.uk

1 Introduction

Plotting a peristimulus time histogram (PSTH), or a spike density function (SDF), from spiketrains evoked by and aligned to the onset of a stimulus is often one of the first steps in the analysis of neurophysiological data. It is an easy way of visualising certain characteristics of the neural response, such as instantaneous firing rates (or firing probabilities), latencies and response offsets. These measures also implicitly represent a model of the neuron's response as a function of time and are important parts of their functional description. Yet PSTHs are frequently constructed in an unsystematic manner, e.g. the choice of time bin size is driven by result expectations as much as by the data. Recently, there have been more principled approaches to the problem of determining the appropriate temporal resolution (Shimazaki and Shinomoto 2007a, b, c).

We recently developed an exact Bayesian, generative model approach to estimating PSTH/SDFs (Endres et al. 2008). Our model encodes a spike generator described by an inhomogeneous Bernoulli process with piecewise constant (in time) firing probabilities. We demonstrated that relevant marginal distributions, e.g. the posterior distribution of the number of bins, can be evaluated from the full posterior distribution over the model parameters efficiently, i.e. in polynomial time. Extending earlier dynamic programming schemes

Published online: 16 May 2009

(Endres and Földiák 2005), we also showed that expected values, such as the predictive firing rate and its standard error, are computable with at most cubic effort.

Here we extend the performance comparisons in (Endres et al. 2008) and illustrate the usefulness of our method. We also demonstrate how to use our Bayesian approach for principled feature extraction from spike trains. Specifically we examine latencies and firing rates, since previous studies (Richmond and Optican 1987b; Tovee et al. 1993) indicate that much of the stimulus-related information carried by neurons is contained in these measures (see Oram et al. 2002 for a review). We give a 'minimal' definition of latency and show how the latency posterior distribution and the firing rate posterior density can be evaluated. These posteriors are then employed for an information theoretic analysis of the neural code comprised of latency and firing rate. Note that we do in no way claim that a PSTH is a complete generative description of spiking neurons. We are merely concerned with inferring that part of the generative process which can be described by a PSTH in a Bayes-optimal way. This paper tries to appeal to computational neuroscientists and neurophysiologists alike. While the former require sound derivations to accept a method's validity, the latter need to be convinced of a method's superiority through demonstrations if they are to adopt it. We attempt to present a balanced mix of both.

2 The model

2.1 Traditional approaches

The traditional approaches to estimating firing probabilities or firing rates from neurophysiological data can roughly be divided into two classes: binning and smoothing. The former yields PSTHs, whereas the latter produces SDFs (Richmond and Optican 1987a). Both are instances of regularisation procedures, which try to deal with the ubiquitous noise and data scarcity by making various implicit assumptions. From a generative model perspective, binning basically presupposes that the firing probabilities are constant within each bin, whereas smoothing imposes the prior belief that high-frequency fluctuations are mostly noise. Whether these assumptions are correct can not be decided a priori, but must be evaluated by comparing the predictive performances of all models in question on real neurophysiological data (see Section 3.4).

An intuitive understanding of the relative merits and drawbacks of these two approaches can be obtained from Fig. 1: panel (a) shows a rastergram of 32 spiketrains recorded from an STSa neuron in response to a stimulus. Each tick represents a spike, with the spiketrains (rows) aligned to stimulus onset. Panel (b) shows a PSTH with a fixed bin duration, optimised for the data by the method described in Shimazaki and Shinomoto (2007b, c). While a bin PSTH could in principle model sharp transients, the location of the bin boundaries are determined by the constant binwidth. Therefore, the precise onset of the transient is often not captured well. In addition, the constant bin duration also forces this method to put many bins into time intervals where the spiketrains appears relatively constant, e.g. in [200 ms, 400 ms]. Panel (c) depicts the SDF obtained by smoothing these spiketrains with a Gaussian

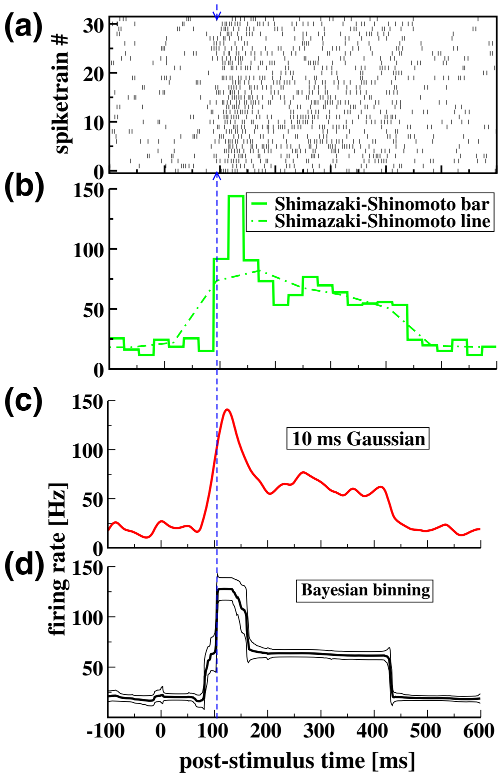

Fig. 1 Predicting a PSTH/SDF with 3 different methods. (**a**) the dataset used in this comparison consisted of 32 spiketrains recorded from a STSa neuron. This neuron was chosen for its clear response profile. Each tick mark represents a spike. (**b**) bar PSTH (*solid line*), optimal binsize ≈26 ms, and line PSTH (*dashed line*), optimal binsize ≈78 ms, computed by the methods described in Shimazaki and Shinomoto (2007b, c). (**c**) SDF obtained by smoothing the spike trains with a 10 ms Gaussian kernel. (**d**): PSTH inferred with our Bayesian binning method. The *thick line* represents the predictive firing rate, the *thin lines* show the predictive firing rate ±1 standard deviation. Models with $4 \leq M \leq 13$ were included on a risk level of $\alpha = 0.1$ (see Eq. (17)).The *vertical dashed line* indicates the mode of the latency posterior (see Section 4.1 and Fig. 5)

kernel of 10 ms width. Compared to the rastergram, high frequency fluctuation in the spiketrains is reduced to some degree, as can be seen e.g. in the interval [200 ms, 400 ms]. However, the sharp transient at ≈100 ms (indicated by the dashed vertical line across panels (b–d)), becomes blurred. Thus, relevant timing information might be lost.

Finally, both binning and smoothing are often employed to compute point estimates of the instantaneous firing rate. Given the typically small sample sizes in neurophysiological experiments, reliable point estimates are hard to obtain, and measures of posterior uncertainty and variability, both between and within trials, should be a part of the estimation procedure. Our Bayesian binning method (Fig. 1(d)) achieves this goal.

2.2 Bayesian binning

We propose a compromise between binning and smoothing to deal with the problems described in the previous section: keep the bins to allow for rapid changes in the instantaneous firing rate, but allow for varying bin durations. This enables us to put the bin boundaries at only those time points where the changes in firing rate happen. As a consequence, time intervals in which the firing rate does not change can now be modelled by one (or a few) bins, which reduces the risk of overfitting noise. Uncertainties and variabilities will be computed in an exact Bayesian fashion. The resultant expected firing rates (complete with their uncertainties) will therefore have a more continuous appearance, similar to the results yielded by a smoothing technique.

Details of the formal model have been described in Endres et al. (2008). Briefly, we model a PSTH on $[t_{\min}, t_{\max}]$ discretised into T contiguous intervals of duration $\Delta t = (t_{\max} - t_{\min})/T$ (see Fig. 2(a and b)). We select a discretisation fine enough (here 1 ms) so that we will not observe more than one spike in a Δt interval for any given spike train. Spike train i can then be represented by a binary vector $\vec{z}^i$ of dimensionality T. We model the PSTH by $M + 1$ contiguous, non-overlapping bins having inclusive upper boundaries k_m, within which the firing probability $f_m = P(\text{spike}|t \in (t_{\min} + \Delta t(k_{m-1} + 1), t_{\min} + \Delta t(k_m + 1)])$ is constant. Importantly, the bin size (distance between bin boundaries) is not fixed *a priori* but can vary depending on the observed data. The relationship between the firing probabilities f_m and the instantaneous firing rates is given by

$$\text{firing rate} = \frac{f_m}{\Delta t}. \tag{1}$$

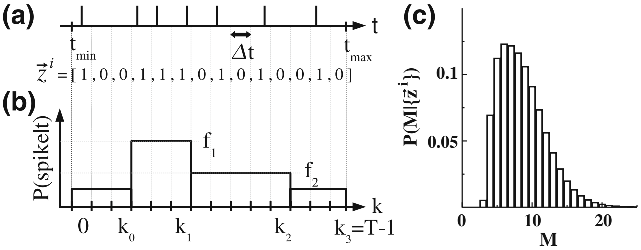

Fig. 2 (**a**) A spike train, recorded between times $t_{\min}$ and $t_{\max}$ is represented by a binary vector $\vec{z}^i$. (**b**) The time span between $t_{\min}$ and $t_{\max}$ is discretised into T intervals of duration $\Delta t = (t_{\max} - t_{\min})/T$, such that interval k lasts from $k \times \Delta t + t_{\min}$ to $(k + 1) \times \Delta t + t_{\min}$. Δt is chosen such that at most one spike is observed per Δt interval for any given spike train. Then, we model the firing probabilities $P(\text{spike}|t)$ by $M + 1 = 4$ contiguous, non-overlapping bins (M is the number of bin boundaries inside the time span $[t_{\min}, t_{\max}]$), having inclusive upper boundaries k_m and $P(\text{spike}|t \in (t_{\min} + \Delta t(k_{m-1} + 1), t_{\min} + \Delta t(k_m + 1)]) = f_m$. (**c**) model posterior $P(M|\{\vec{z}^i\})$ (see Eq. (16)) computed from the data shown in Fig. 1. The shape is fairly typical for model posteriors computed from the neural data used in this paper: a sharp rise at a moderately low M followed by a maximum (here at $M = 6$) and an approximately exponential decay. Even though a maximum M of 699 would have been possible, $P(M > 23|\{\vec{z}^i\}) < 0.001$. Thus, we can accelerate the averaging process for quantities of interest (e.g. the predictive firing rate) by choosing a moderately small maximum M. For details, see text

M is the number of bin boundaries inside $[t_{\min}, t_{\max}]$. The probability of a spike train $\vec{z}^i$ of independent spikes/gaps is then

$$P(\vec{z}^i|\{f_m\}, \{k_m\}, M) = \prod_{m=0}^{M} f_m^{s(\vec{z}^i,m)}(1 - f_m)^{g(\vec{z}^i,m)} \tag{2}$$

where $s(\vec{z}^i, m)$ is the number of spikes and $g(\vec{z}^i, m)$ is the number of non-spikes, or gaps in spiketrain $\vec{z}^i$ in bin m, i.e. between intervals $k_{m-1} + 1$ and k_m (both inclusive). In other words, we model the spiketrains by an inhomogeneous Bernoulli process with piecewise constant probabilities. We also define $k_{-1} = -1$ and $k_M = T - 1$. Note that there is no binomial factor associated with the contribution of each bin, because we do *not* want to ignore the spike timing information within the bins, but rather, we try to build a simplified generative model of the spike train. Therefore, the probability of a (multi) set of spiketrains $\{\vec{z}^i\} = \{z_1, \ldots, z_N\}$, assuming independent generation, is

$$\begin{aligned} P(\{\vec{z}^i\}|\{f_m\}, \{k_m\}, M) &= \prod_{i=1}^{N}\prod_{m=0}^{M} f_m^{s(\vec{z}^i,m)}(1 - f_m)^{g(\vec{z}^i,m)} \\ &= \prod_{m=0}^{M} f_m^{s(\{\vec{z}^i\},m)}(1 - f_m)^{g(\{\vec{z}^i\},m)} \end{aligned} \tag{3}$$

where $s(\{\vec{z}^i\}, m) = \sum_{i=1}^N s(\vec{z}^i, m)$ and $g(\{\vec{z}^i\}, m) = \sum_{i=1}^N g(\vec{z}^i, m)$.

2.3 The priors

We make a non-informative prior assumption for the joint prior of the firing probabilities $\{f_m\}$ and the bin boundaries $\{k_m\}$ given the total number of bin boundaries M, namely

$$p(\{f_m\}, \{k_m\}|M) = p(\{f_m\}|M)P(\{k_m\}|M). \tag{4}$$

i.e. we have no a priori preferences for the firing rates based on the bin boundary positions. Note that the prior of the f_m, being continuous model parameters, is a density. Given the form of Eq. (2) and the constraint $f_m \in [0, 1]$, it is natural to choose a conjugate prior

$$p(\{f_m\}|M) = \prod_{m=0}^{M} \mathrm{B}(f_m; \sigma_m, \gamma_m). \tag{5}$$

The Beta density is defined in the usual way (see e.g. Berger 1985):

$$\mathrm{B}(p; \sigma, \gamma) = \frac{\Gamma(\sigma+\gamma)}{\Gamma(\sigma)\Gamma(\gamma)} p^{\sigma-1}(1-p)^{\gamma-1}. \tag{6}$$

There are only finitely many configurations of the k_m. Assuming we have no preferences for any of them, the prior for the bin boundaries becomes

$$P(\{k_m\}|M) = \frac{1}{\binom{T-1}{M}}. \tag{7}$$

where the denominator is just the number of possibilities in which M ordered bin boundaries can be distributed across $T-1$ places (bin boundary M always occupies position $T-1$, see Fig. 2(b), hence there are only $T-1$ positions left).

2.4 Computing the evidence and other posterior expectations

To calculate quantities of interest for a given number of bin boundaries M and a set of spiketrains $\{\vec{z}^i\}$, e.g. predicted firing probabilities, their variances and expected bin boundary positions, we need to average the quantity of interest over the posterior of the firing rates in the bins $\{f_m\}$ and the bin boundaries $\{k_m\}$:

$$p(\{f_m\}, \{k_m\}|M, \{\vec{z}^i\}) = \frac{p(\{\vec{z}^i\}, \{f_m\}, \{k_m\}|M)}{P(\{\vec{z}^i\}|M)} \tag{8}$$

which requires the evaluation of the evidence, or marginal likelihood of a model with M bins:

$$P(\{\vec{z}^i\}|M) = \sum_{k_{M-1}=M-1}^{T-2} \sum_{k_{M-2}=M-2}^{k_{M-1}-1} \dots \sum_{k_0=0}^{k_1-1} P(\{\vec{z}^i\}|\{k_m\}, M)P(\{k_m\}|M) \tag{9}$$

where the summation boundaries are chosen such that the bins are non-overlapping and contiguous and

$$P(\{\vec{z}^i\}|\{k_m\}, M) = \int_0^1 d\{f_m\} P(\{\vec{z}^i\}|\{f_m\}, \{k_m\}, M) p(\{f_m\}|M). \tag{10}$$

with

$$\int_0^1 d\{f_m\} = \int_0^1 df_0 \int_0^1 df_1 \dots \int_0^1 df_M. \tag{11}$$

Computing the sums in Eq. (9) might seem difficult. M sums over $O(T)$ summands suggest a computational complexity of $O(T^M)$, which is impractical. To appreciate why, let's consider an example: In a typical neurophysiological scenario, we might want to estimate the PSTH in a $T = 700$ ms time window with $\Delta t = 1$ ms. If we tried to model this distribution by $M + 1 = 11$ bins, we would have to check $\binom{699}{10}$ configurations, i.e. the number of possibilities to distribute 10 ordered bin boundaries across 699 places. This is $> 10^{21}$. Even if we checked 10 configurations per microsecond, we would take more than 20 million years to finish.

However, we can expedite this process. As previously demonstrated (Endres et al. 2008), using dynamic programming the computational complexity can be reduced to $O(MT^2)$. In the above example, the time to compute the evidence reduces to ≈0.5 s, which is fast enough to be useful. We give a description of the algorithm in Appendix A. This algorithm is also the basis for the latency calculations in Section 4.1.

Posterior expectations can be evaluated in a similar fashion. For example, given the model parameters $\{k_m\}$,$\{f_m\}$ and M, the predictive firing probability at time index t can formally be written as

$$P(\text{spike}|t, \{f_m\}, \{k_m\}, M) = \sum_{m=0}^{M} f_m \mathscr{T}(t \in \{k_{m-1}+1, k_m\}) \tag{12}$$

where the indicator function $\mathscr{T}(x) = 1$ iff x is true and 0 otherwise. Thus, the sum will have exactly one nonzero contribution from that bin which contains t. Multiplying the r.h.s. of Eq. (12) with the r.h.s. Eq. (8)

and marginalising $\{f_m\}$ and $\{k_m\}$ yields the predictive firing probability at t given M and the data $\{\vec{z}^i\}$:

$$\left\langle P(\text{spike}|t)\right\rangle \tag{13}$$

where $\langle\ldots\rangle$ denotes a posterior expectation. The necessary summations/integrations can be done by a modified version of the algorithm described in Appendix A: since Eq. (12) puts a factor f_m into the bin which contains t, we only need to add an 'extra' spike in this bin in Eq. (35), run the algorithm and divide the result by the evidence to obtain the predictive firing probability.

To compute the standard deviation of the firing probability, we need the posterior expectation of

$$P^2(\text{spike}|t,\{f_m\},\{k_m\},M)=\sum_{m=0}^{M} f_m^2\mathscr{T}(t\in\{k_{m-1}+1,k_m\}) \tag{14}$$

The factor f_m^2 amounts to putting *two* spikes in the bin which contains t. Then,

$$\text{Var}\left(P(\text{spike}|t)\right)=\left\langle P^2(\text{spike}|t,\{\vec{z}^i\},M)\right\rangle - \left\langle P(\text{spike}|t,\{\vec{z}^i\},M)^2\right\rangle \tag{15}$$

2.5 Model selection vs. model averaging: how many bins do we need?

To choose the best M given $\{\vec{z}^i\}$, or better, a probable range of Ms, we need to determine the model posterior

$$P(M|\{\vec{z}^i\})=\frac{P(\{\vec{z}^i\}|M)P(M)}{\sum_m P(\{\vec{z}^i\}|m)P(m)} \tag{16}$$

where $P(M)$ is the prior over M, which we assume to be uniform. The motivation for this choice is simply that we have no a priori preferences for any model complexity, but we would rather drive the choice of M as completely as possible by the data. The sum in the denominator runs over all values of m which we choose to include, at most $m\leq T-1$.

Once $P(M|\{\vec{z}^i\})$ is evaluated, we could use it to select the most probable M'. However, making this decision means 'contriving' information, namely that all of the posterior probability is concentrated at M'. Thus we should rather average any predictions over all possible M, even if evaluating such an average has a computational cost of $O(T^3)$, since $M\leq T-1$. If the structure of the data allow, it is possible, and useful given a large enough T, to reduce this cost by finding a range of M, such that the risk of excluding a model even though it provides a good description of the data is low. In analogy to the significance levels of orthodox statistics, we shall call this risk α. If the posterior of M is unimodal (which it has been in most observed cases, see Fig. 2(c), for an example), we can then choose the smallest interval of Ms around the maximum of $P(M|\{\vec{z}^i\})$ such that

$$P(M_{\min}\leq M\leq M_{\max}|\{\vec{z}^i\})\leq 1-\alpha \tag{17}$$

and carry out the averages over this range of M after renormalising the model posterior. We use $\alpha=0.1$ unless stated otherwise.

3 Simulations and comparison to other methods

3.1 Predicted PSTH convergence to simulated generator

We first tested our method by inferring PSTH/SDFs from artificial data. We generated spiketrains from inhomogeneous Bernoulli processes with the rate profiles shown in the top panels of Fig. 3. To quantify the difference between the generator and an inferred PSTH/SDF, we employed a time-averaged version of the Kullback-Leibler divergence (KLd) (Cover and Thomas 1991). Let $P(t)$ and $Q(t)$ be the spiking probability of the generator and the inferred PSTH/SDF at time t, respectively. The KLd between them at t is

$$\text{KLd}(t)=P(t)\log\left(\frac{P(t)}{Q(t)}\right)+(1-P(t))\log\left(\frac{1-P(t)}{1-Q(t)}\right). \tag{18}$$

KLd has several interpretations, the one most relevant for our purposes is the following: if we had observed a spike at time t, $\log\left(\frac{P(t)}{Q(t)}\right)=\log\left(P(t)\right)-\log\left(Q(t)\right)$ would measure how much more (log-) probable that spike would have been given the generator versus the inferred PSTH/SDF. Likewise, if we had observed no spike, $\log\left(\frac{1-P(t)}{1-Q(t)}\right)$ tells us how much more (log-) probable this event would have been. To get the expected gain in (log-)probability, we need to average these terms over the spike/no spike generating distribution at t, which is given by $P(t)$ and $1-P(t)$, respectively. This averaging yields Eq. (18). It can be shown (Cover and Thomas 1991) that KLd≥ 0 with equality only if $P(t)=Q(t)$, i.e. the expected log probability of spike/no spike is maximised by the generating distribution. We

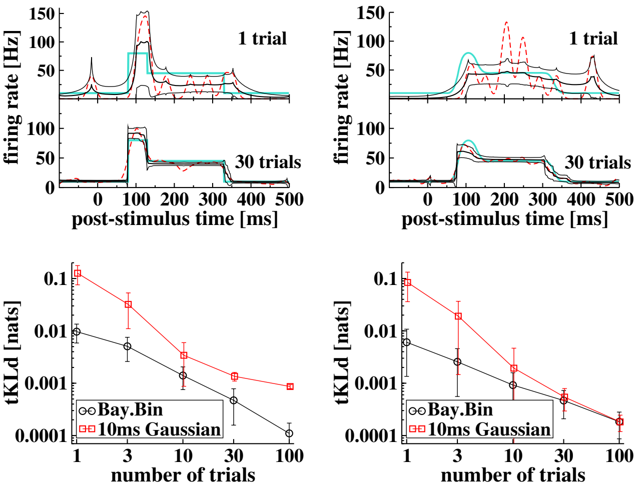

Fig. 3 Performance comparsion on artificially generated spiketrains. The generators (thick, square-wave (*left*) and smoothed square-wave (*right*) lines in *top panels*) are the rate profiles from which the spike trains were drawn. *Top panels* show typical PSTH/SDFs obtained from datasets containing 1 and 30 trials. 'Typical' means that the time-averaged Kullback-Leibler divergence (tKLd) between the generator and the estimated PSTH/SDFs is close to the average tKLd for a given number of trials. *Dashed*: smoothing with a Gaussian kernel of 10 ms width, *Solid*: Bayesian binning. *Bottom panels*: average tKLd between generator and estimated PSTH/SDFs across 100 simulations as a function of trials per dataset. The generator on the left is comprised of bins, which Bayesian binning should be able to model perfectly given a large enough dataset size. Thus, the tKLd at 100 trials is much smaller for Bayesian binning. More importantly, Bayesian binning is consistently better than Gaussian smoothing even for very small numbers of trials. The generator on the right is smoothed with a 10 ms wide Gaussian kernel. While Bayesian binning can no longer model it perfectly with a finite number of bins, it is still a better estimator than kernel smoothing up to at least 100 trials

average KLd(t) across all time indexes of interest to yield the time-averaged KLd (tKld):

$$\text{tKLd} = \frac{1}{T}\sum_{t=0}^{T-1}\text{tKLd}(t) \tag{19}$$

The top panels in Fig. 3 show typical PSTH/SDFs inferred from 1 and 30 trials. 'Typical' means that the tKLd is close to the average tKLd for a given number of trials. The Bayesian binning PSTH is computed from the predictive firing probability $\langle P(\text{spike}|t)\rangle$, the dashed lines represent ±1 posterior standard deviation (from Eq. (15)), the prior parameters σ_m and γ_m were equal for all bins and set to their maximum a-posteriori value. The generating rate profile in the left half of Fig. 3 is comprised of bins. Hence, Bayesian binning should model it with increasing accuracy and reduced uncertainty as the dataset grows. An indication for that can be seen by comparing the PSTH/SDFs from 1 and 30 trials: the generating rate profile is followed much more closely for 30 trials than for 1 and the posterior standard deviations also decrease noticeably as the number of trials increases. Furthermore, the Bayesian binning PSTH is closer to the generator than the SDF computed by smoothing the spiketrains with a 10 ms wide Gaussian kernel, which is displayed for comparison.

Importantly, Bayesian binning is doing well even if the generator cannot be modelled by a small number of bins: the right half of Fig. 3 shows simulation results for a generator that was smoothed with a 10 ms wide Gaussian kernel. Here, the Gaussian kernel smoothing gives expectably good results (at least for 30 trials), but note that Bayesian binning is doing apparently equally as well. More quantitative performance comparison results are shown in the bottom panels of Fig. 3. We repeated the simulation 100 times for a given number of trials per dataset, thus obtaining the average tKLd and its standard deviation. For the bin generator, Bayesian binning outperforms Gaussian kernel smoothing for all dataset sizes. For the smoothed generator, Bayesian binning still outperforms Gaussian kernel smoothing, while the difference between the two methods shrinks as the number of trials per dataset increases. But even for 100 trials, Bayesian binning is as good as Gaussian kernel smoothing. We have thus reason to hope that Bayesian binning might outperform other PSTH/SDF estimation methods on real neural data. This will be shown in the next subsections.

3.2 Data acquisition

The experimental protocols have been described before (Oram et al. 2002; van Rossum et al. 2008). Briefly, extra-cellular single-unit recordings were made using standard techniques from the upper and lower banks of the anterior part of the superior temporal sulcus (STSa) of two monkeys (Macaca mulatta) performing a visual fixation task. The subject received a drop of fruit juice reward every 500 ms of fixation while static stimuli (10° by 12.5°) were displayed. Static images were presented centrally on the monitor. Stimuli consisted of 256 gray scale pictures of familiar and unfamiliar objects, heads, body parts and whole bodies. Visual stimuli were presented in a random sequence for 333 ms with a 333 ms inter-stimulus interval centrally on a black monitor screen (Sony GDM-20D11, resolution 25.7 pixels/degree, refresh rate 72 Hz), 57 cm from the subject. Stimulus contrast was determined using foreground regions of the image. The 100% Michelson contrast = $\frac{L_{\max}-L_{\min}}{L_{\max}+L_{\min}}$, where L is the luminance, was formed by normalising the foreground pixel values such that they occupied the monitor full luminance range after adjusting the initial greyscale image to have mid (50%) luminance. Other contrast versions (75%, 50%, 25%, 12.5%) were achieved by systematically varying the width of the distribution of the foreground pixel values of the 100% contrast version while maintaining the average foreground luminance. All manipulations were performed after correcting for the measured gamma function of the display monitor.

Stimulus presentation began after 500 ms of fixation centrally on the screen (fixation deviations outside the fixation window lasting ≤100 ms were ignored to allow for blinking). Fixation was rewarded with the delivery of fruit juice. Spikes were recorded during the period of fixation. If the subject looked away for longer than 100 ms, both spike recording and presentation of stimuli stopped until the subject resumed fixation for 500 ms. The results from initial screening (Edwards et al. 2003) were used to select stimuli that elicited large responses from the neuron (effective stimuli) and to select stimuli that elicited small or no response (ineffective stimuli). For different neurons effective and ineffective stimuli included different views of the head (Perrett et al. 1991), abstract patterns and familiar objects (Földiák et al. 2004). Details of the stimulus selectivity of these neurons has been reported elsewhere (Oram et al. 2002; Földiák et al. 2004; Edwards et al. 2003; Barraclough et al. 2005). The anterior–posterior extent of the recorded cells was from 7 mm to 10 mm anterior of the interaural plane, in the upper bank (TAa, TPO), lower bank (TEa, TEm) and fundus (PGa, IPa) of the superior temporal sulcus (STS) and in the anterior areas of TE (AIT of [Tanaka1991]), areas which we collectively call the anterior STS (STSa, see Barraclough et al. (2005) for further discussion). The recorded firing patterns were turned into distinct samples, each of which contained the spikes from −300 ms to 600 ms after the stimulus onset with a temporal resolution of 1 ms.

3.3 Inferring PSTHs

To see the method in action on real neural data, we inferred a PSTH from 32 spiketrains recorded from one of the available STSa neurons (see Fig. 1(a)). We discretised the interval from −100 ms pre-stimulus to 600 ms post-stimulus into $\Delta t = 1$ ms time intervals and computed the posterior (16) for models with varying number of bins M (see Fig. 2(c)). The prior parameters were equal for all bins and set to $\sigma_m = 1$ and $\gamma_m = 32$. This choice corresponds to a firing probability of ≈0.03 in each 1 ms time interval (30 spikes/s), which is typical for the neurons in this study.[1] Models with $4 \leq M \leq 13$ (expected bin sizes between ≈23 ms–148 ms) were included on an $\alpha = 0.1$ risk level (Eq. (17)) in the subsequent calculation of the predictive firing rate (i.e. the *expected* firing rate, hence the continuous appearance) and standard deviation (Fig. 1(d)). For comparison, Fig. 1(b), shows a bar PSTH and a line PSTH computed with the recently developed methods described in Shimazaki and Shinomoto (2007b, c). Roughly speaking, these methods try to optimise a compromise between minimal within-bin variance and maximal between-bin variance. In this example, the bar PSTH consists of 26 bins. Panel (c) in Fig. 1 depicts a SDF obtained by smoothing the spiketrains with a 10 ms wide Gaussian kernel, a standard way of calculating SDFs in the neurophysiological literature.

All tested methods produce results which are, upon cursory visual inspection, largely consistent with the spiketrains. However, Bayesian binning is better suited than Gaussian smoothing to model steep changes, such as the transient response starting at ≈100 ms. While the methods from Shimazaki and Shinomoto (2007b, c) share this advantage, they suffer from two drawbacks: firstly, the bin boundaries are evenly spaced, hence the peak of the transient is later than visual examination of the rastergrams would suggest. Secondly, because the bin duration is the only parameter of the model, these methods are forced to put many bins even in intervals that are relatively constant, such as the baselines before and after the stimulus-driven response. In contrast, Bayesian binning is able to put bin boundaries anywhere in the time span of interest and can model the data with less bins—the model posterior has its maximum at $M = 6$ (7 bins), whereas the bar PSTH consists of 26 bins.

3.4 Performance comparison by cross-validation

For a more rigorous method comparison, we split the data into distinct sets, each of which contained the responses of a cell to a different stimulus. This procedure yielded 336 sets from 20 cells with at least 20 spiketrains per set. We then performed 5-fold crossvalidation. The crossvalidation error is given by the negative logarithm of the predicted probability (Eq. (13)) of the data (spike or no spike) in the test sets. Let $s_n(t) = 1$ if trial n of N in the

[1] Alternatively, one could search for the σ_m, γ_m which maximise of $P(\{\vec{z}^i\}|\sigma_m, \gamma_m) = \sum_M P(\{\vec{z}^i\}|M)P(M|\sigma_m, \gamma_m)$, where $P(\{\vec{z}^i\}|M)$ is given by Eq. (9). Using a uniform $P(M|\sigma_m, \gamma_m)$, we found $\sigma_m \approx 2.3$ and $\gamma_m \approx 37$ for the data in Fig. 1(a).

test set contains a spike at time index $t \in \{0, \dots, T-1\}$ and $s_n(t) = 0$ otherwise. Then

$$\text{CV error} = -\frac{1}{N}\sum_{n=0}^{N-1}\frac{1}{T}\sum_{t=0}^{T-1}\log\langle(P(s_n(t)|t))\rangle\,. \tag{20}$$

Thus, we measure how well the PSTHs/SDFs predict the test data on average across time and across all test trials. Note that this CV error is similar to the tKLd (Eq. (19)): the constant terms referring to the generator have been dropped, because the generator is not known here and the averaging is done across the data rather than the generating distribution for the same reason. We average the CV error over the 5 estimates to obtain a single estimate for each of the 336 neuron/stimulus combinations. The prior parameters σ_m, γ_m were equal for all bins and MAP optimised for each individual training dataset. In Endres et al. (2008) we already demonstrated that Bayesian binning outperforms SDFs obtained by Gaussian smoothing, and the bin and line histogram methods from Shimazaki and Shinomoto (2007b, c).

We also tested Bayesian binning against the kernel smoothing method described in (Shimazaki and Shinomoto 2007a), a local likelihood adaptive fit (Loader 1999) and Bayesian Adaptive Regression Splines (BARS) (DiMatteo et al. 2001). To compare the performances between the different methods directly, we calculated the difference in CV error for each neuron/stimulus configuration. Here a positive value indicates that Bayesian binning predicts the test data more accurately than the alternative method. Figure 4, shows the relative frequencies of CV error differences between the other methods and our approach. In the large majority of cases we are at least as good, but frequently better than the competitors, indicating the general utility of our approach. Amongst the competitors, BARS is the only method with a comparable predictive performance on these STSa data. The average CV error differences, summarised in Table 1, support this claim: they are all significantly > 0, except for the BARS value.

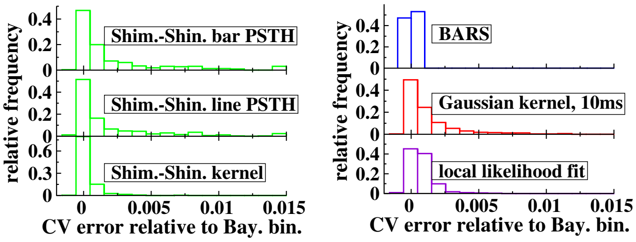

Fig. 4 Comparison of Bayesian Binning with competing methods by 5-fold crossvalidation. The CV error is the negative expected log-probability of the test data. The histograms show relative frequencies of CV error differences to our Bayesian binning approach. *Left*: Shimazaki's and Shinomoto's methods (Shimazaki and Shinomoto 2007a, b). *Right, top* Bayesian Adaptive Regression Splines (*BARS*) (DiMatteo et al. 2001). *Right, middle*: smoothing with a Gaussian kernel of 10 ms width. *Right, bottom*: local likelihood adaptive fitting (Loader 1997, 1999)

Table 1 Average log prediction error differences to Bayesian binning from 5 fold crossvalidation on 336 datasets

Method	CV error diff.
Shimazaki and Shinomoto (2007b) bar	$(2.35 \pm 0.23) \times 10^{-3}$
Shimazaki and Shinomoto (2007b) line	$(1.22 \pm 0.10) \times 10^{-3}$
Gauss 10 ms	$(1.29 \pm 0.11) \times 10^{-3}$
Local likelihood fit (Loader 1997)	$(7.34 \pm 0.48) \times 10^{-4}$
Shimazaki and Shinomoto (2007a) kernel	$(3.14 \pm 0.39) \times 10^{-4}$
BARS (DiMatteo et al. 2001)	$(0.8 \pm 1.6) \times 10^{-5}$
Bayesian binning	0

A positive value means that our method predicts the data better than the competitor

4 Response latency

Besides the instantaneous firing rate, another frequently used feature for the description of a neuron's response is response latency. But unlike the former, a definition of latency seems much less agreed. A wide range of methods to estimate response latency exist. Changes in phase between neuronal activity and sinusoidal drifting gratings with changing stimulus parameters can provide an indirect measure of response latency (Gawne et al. 1996b; Alitto and Usrey 2004). Direct measures of response latency of neurons with low background or spontaneous activity can be obtained from the time of the first spike after stimulus onset (Heil and Irvine 1997; Richmond et al. 1999; Syka et al. 2000; Stecker and Middlebrooks 2003; Hurley and Pollak 2005).

Statistical approaches compare activity levels at two time points. While the baseline level is usually taken from a "pre-stimulus" period the window containing the greatest activity can be used as the reference point (Berenyi et al. 2007). Comparison of the baseline or reference activity with the activity in a sliding window using t-tests (Sugase-Miyamoto and Richmond 2005; Berenyi et al. 2007) can be used to determine the time point at which neuronal activity changes and hence provide an estimate of response latency.

Several approaches use either the PSTH or the SDF to determine neuronal response latency. Latency estimates can be based on peak activity, typically the time at which the mean activity reaches half the amplitude of the peak (baseline + 0.5 × (peak − baseline), e.g. (Gawne et al. 1996a; Lee et al. 2007)). A statistical method based on a Poisson model compares the mean activity in successive bins during stimulation with a Poisson process estimated from the "pre-stimulus" period (Maunsell and Gibson 1992; Nowak et al. 1995; Hanes et al. 1995; Thompson et al. 1996; Schmolesky et al. 2006; Gabel et al. 2002; Sary et al. 2006). However, Friedman and Priebe (1998) concluded that a maximum likelihood estimation of parameters for a step change in Poissonian generator (rate 1 pre-latency, rate 2 post-latency) was a better methodology in terms of mean square error than using half-height (Gawne et al. 1996a) and the Poisson assumption approach (Maunsell and Gibson 1992).

Some statistical approaches to estimating response latency use measures of the variability obtained from the data rather than assume Poisson statistics. Simple methods estimate response latency as the time point at which activity exceeds baseline plus some error margin (e.g. 1.96 or 2.58 standard error of mean (SEM) of baseline, (Oram and Perrett 1992; Oram and Perret 1996; Tamura and Tanaka 2001; Edwards et al. 2003; Eifuku et al. 2004; Kiani et al. 2005; van Rossum et al. 2008). Such thresholding can also determine if a visually induced response is present (e.g. baseline+3.72 SEM, (Lee et al. 2007)). Of course, estimates of latency derived from the SDF will vary with the width of the smoothing kernel. Ingenious methods involving estimates from multiple kernels of different widths have been developed to minimise this effect (Liu and Richmond 2000).

Other methods developed to estimate response latency include using ROC analysis of single cell recordings (Tanaka and Lisberger 2002). Estimating response latency as the time of the peak in the derivative of the SDF from multi-unit and local field potential recordings (Fries et al. 2001) relies on rapid change in firing rate at response onset. Taking the first time bin of the longest monotonic rise in activity (Liu and Richmond 2000) relies on a large, but not necessarily fast, change in activity. Finally, Luczak and colleagues (Luczak et al. 2007) use the mean spike time after stimulus onset as a latency measure.

Methods have also been developed that allow for estimation of the response latency of a single trial. Some calculate the trial-by-trial variability of response latency but do not give the absolute latency (Nawrot et al. 2003). Other statistical approaches, including the Poisson based methods (Maunsell and Gibson 1992; Hanes et al. 1995; Thompson et al. 1996; Sary et al. 2006) and the "baseline+error margin" methods, can provide latency estimates for single trials although they may not return a latency estimate for every trial (Friedman and Priebe 1998). The trial alignment approach from Ventura (2004) builds on the observation that a PSTH, when normalised across time, can be interpreted as a probability distribution for generating spike times. Assuming that the shape of the PSTH does not change across trials, but may be shifted in time relative to other trials, the difference between trial latencies must then be equal to the difference of mean spike times. To compute an absolute latency, Ventura (2004) recommends to align all trials to the minimal trial mean and use a point estimation method on the aligned trials, since the alignment should facilitate the detection of a sharp onset. Confidence intervals on the latency estimates can be obtained via bootstrap.

We note that most of the methods listed above share the notion of determining latency by estimating a point value. However, with finite data there is always uncertainty in the estimate. For example, when latency is estimated as 100 ms it could be 99 ms or 101 ms with almost as much certainty but is relatively unlikely to be 90 or 110 ms. If we want to search for patterns or changes in response latency more exacting analysis techniques should thus incorporate the uncertainty in a principled fashion. We also want a single method that, without any change to parameters or code, works with individual trials, with a set of trials to a single stimulus and with all trials from a neuron. We now develop and evaluate latency estimation using our Bayesian binning technique and show it meets these two criteria.

4.1 A minimal definition of response latency

Most people interested in latency would probably agree with the notion that 'latency is where the signal starts'. Signal vs. no signal can usually be translated into firing rate above or below a threshold, which we will call the *signal level* (see Fig. 5, left). In other words, *latency is that point in time prior to which there was no signal, and after which there is a signal for at least some duration*. This is the 'minimal' latency definition which we will employ in the following.

For given bin boundaries $\{k_m\}$ and firing probabilities $\{f_m\}$, latency must be at a bin boundary, because firing probabilities are constant within each bin. Note that our latency definition implies that there can be at most one latency. If the firing probabilities are below the signal level in every bin, or if f_0, the firing rate in the first bin is already above the signal level, then there will be no latency.

To obtain a latency posterior distribution, we formally define the probability that the latency L is at time index t given $\{k_m\}$,$\{f_m\}$,M and the signal level $S \in [0, 1]$

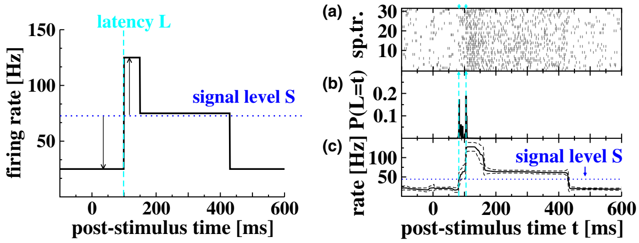

Fig. 5 *Left*: our minimal latency definition. Latency L (*vertical dashed line*) is that point in time before which the firing probability was consistently below the signal level (*dotted horizontal line*), and after which the firing probability is above the signal level for at least one bin. This definition has two important implications: the latency is at a bin boundary, and there can be at most one latency (possibly none). *Right*: (**a**) each tick mark represents a spike, recorded from the same STSa neuron as in Fig. 1 under high-contrast viewing conditions. (**b**) latency posterior. The two modes of $P(L = t)$ are at 83 ms and 104 ms after stimulus onset, indicated by the *dashed vertical lines*. (**c**): expected instantaneous firing rates (*thick solid line*) plus/minus one standard deviation (*thin dashed lines*). This signal level S is indicated by the *horizontal line*. For details, see text

(S is a firing probability. Division by the discretisation stepsize Δt yields firing rate) as

$$P(L = t|\{k_m\}, \{f_m\}, M, S) = \\ = \begin{cases} 1 \text{ if } \exists k_{j-1} \in \{k_m\} : k_{j-1} + 1 = t \\ \quad \text{and } f_j \geq S \text{ and } \forall m < j : f_m < S \\ 0 \text{ otherwise} \end{cases} \tag{21}$$

which can be exactly averaged over the posterior Eq. (8) by a dynamic programming algorithm similar to that used for the evidence evaluation, as detailed in Appendix B. We thus obtain $P(L = t, \{\vec{z}^i\}|M, S)$ and hence, noting that $P(\{\vec{z}^i\}|M) = P(\{\vec{z}^i\}|M, S)$:

$$P(L = t|\{\vec{z}^i\}, M, S) = \frac{P(L = t, \{\vec{z}^i\}|M, S)}{P(\{\vec{z}^i\}|M)}. \tag{22}$$

What remains to be determined is the signal level S. Assuming that the data span the response range of the neuron (i.e. the data contain responses to at least one effective stimulus), one can proceed as follows: for a given S, marginalise the latency posterior across the time interval of interest, thereby obtaining the probability $P(L \text{ exists})$ that a latency exists at that S. Repeat this procedure for different S until the maximal $P(L \text{ exists})$ is found. We use 10 golden section refinement steps (Press et al. 1986) for the maximum search with an initial interval of [0 Hz, 100 Hz], thereby achieving an accuracy of ≤1 Hz.

4.2 Properties of latency posterior distributions

Figure 6 illustrates the consequences of our latency definition on simulated data. We generated 10 spiketrains from inhomogeneous Bernoulli processes with a step in firing rate 10 Hz→80 Hz or 10 Hz→30 Hz at 80 ms after stimulus onset. The firing rate stayed at this value for 50 ms, then dropped to 45 Hz or 20 Hz for 200 ms before returning to the 10 Hz baseline. In both conditions, most of the probability mass of the latency posterior (Fig. 6, left bottom) is concentrated in the vicinity of the generator's latency. The best signal separation level S (Fig. 6, right) for each condition reflects the difference in peak firing rates: for 30 Hz, $S \approx$17 Hz, where as for 80 Hz, $S \approx$39 Hz. In both cases, S is roughly in the middle between baseline and peak firing rate. Latency was searched in the interval [0, 200] ms after stimulus onset.

In addition to the location of the latency, the latency posterior distributions (Fig. 6, left bottom) also contain information about uncertainty. It is evident that a smaller step in firing rate leads to a wider latency posterior, which can also be captured by computing the

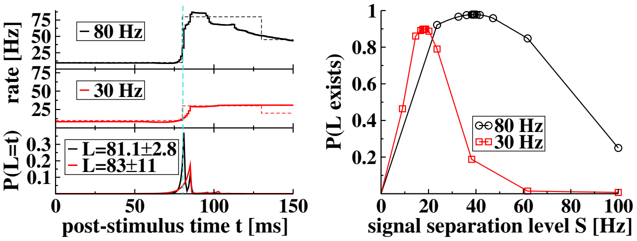

Fig. 6 Latency posterior and signal separation levels. *Left*: 10 spiketrains were drawn from generators with 80 Hz and 30 Hz peak firing rate. Both generators had a baseline of 10 Hz and a latency at 80 ms after stimulus onset. The *dashed lines* show the generating rates, the *solid lines* represent the predictive firing rate of the Bayesian binning PSTHs. The resulting latency posterior distributions are shown at the bottom, including the latency expectations ± 1 posterior standard deviation. The posterior uncertainty in the 30 Hz peak rate condition is significantly larger than in the 80 Hz condition. *Right*: determination of signal separation level S. $P(L \text{ exists})$ is the probability that the latency was somewhere in the latency search interval (here [0, 200] ms after stimulus onset) given S. The symbols are located at the points where $P(L \text{ exists})$ was evaluated by a golden section maximum search (Press et al. 1986). The S was chosen to be the firing rate which maximises the probability that a latency exists. In the 30 Hz condition, there is a relatively clear maximum at ≈17 Hz, whereas in the 80 Hz condition, the maximum is much broader. This is due to the larger difference between baseline and peak firing rate in the latter condition: even for this relatively small dataset (10 trials), there is a range of similarly good signal separation levels that allow for the distinction between baseline and peak firing. For details, see text

standard deviation from that posterior. This observation is not particularly surprising, but nevertheless important: virtually all other latency estimation methods ignore uncertainty due to their point estimation nature. As a consequence, the latency posterior contains information about the change in firing rate, which is a point that we will return to later (Section 5) when we analyse latency and firing rate with information-theoretic methods. Note also that the latency posteriors are far from Gaussian: a description in terms of mean and standard deviation is therefore inadequate for an information-theoretic analysis and might distort conclusions drawn from it.

Non-Gaussian latency posteriors are also observed in the real data. Figure 5, right, has two distinct peaks, the lower one at ≈83 ms, the higher one being at ≈104 ms after stimulus onset. The location of these peaks can be understood from the height of the PSTH (Fig. 5, right, (c)) relative to the signal level: at 83 ms, one can be fairly certain that the PSTH was below the signal level prior to this time index, and there is a nonzero probability (albeit not nearly certainty) that the PSTH is above the signal level directly afterwards. At 104 ms, the PSTH is above the signal level with near certainty directly after the peak in the latency posterior, whereas one can not be quite sure that the PSTH was below the signal level the interval immediately before this point in time. The expected latency ± SEM is (94 ± 10) ms. A conventional interpretation of these values would suggest that the bulk of the probability mass can be found close to the mean, which is not true.

4.3 Simulation results

For a quantitative evaluation of the accuracy of our latency detection method, we generated spiketrains from inhomogeneous Bernoulli processes with the rate profiles shown in the insets of Fig. 7. Root-mean-square (RMS) errors were computed from 100 repetitions of the simulation for a given number of trials per dataset, see Fig. 7. We used the expected latency as the prediction of Bayesian binning for each dataset (similar results were found using a MAP estimate). To further illustrate the performance of out approach, we compared it to three other ways of latency detection: the half-height method (Gawne et al. 1996a) ('HH' in Fig. 7), latency = the first time where activity exceeds baseline rate plus 2 SEM of baseline rate (Oram and Perrett 1992) ('2SD' in Fig. 7) and the trial alignment approach from Ventura (2004). This approach yields a relative latency for each trial, absolute latency can be determined by a suitable change-point method applied to the aligned trials. We used the half-height method here, since it gives good estimates of the latency without alignment.

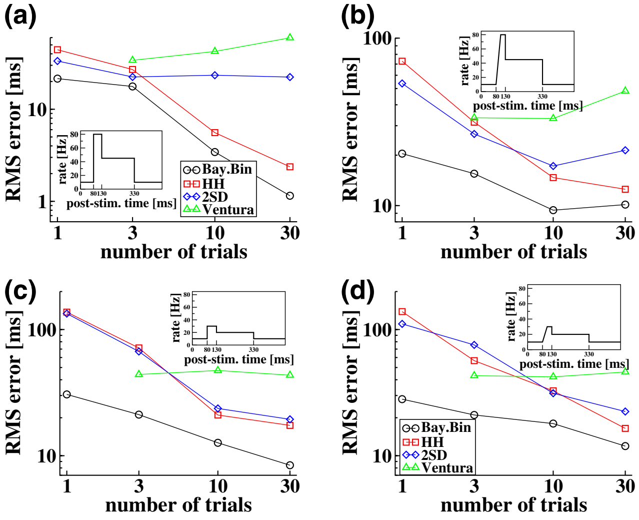

Fig. 7 Comparison of latency estimates. RMS errors were computed from 100 repetitions of the simulation for a given number of trials per dataset. 'HH' are the results from the half-height method (Gawne et al. 1996a), '2SD' determines latency to be the first time where activity exceeds baseline rate plus 2 SEM of baseline rate (Oram and Perrett 1992), 'Ventura' is the trial alignment method from Ventura (2004) and 'Bay.Bin' shows the RMS errors using the expected latency from our method. Insets show generating rate profiles. *Left*: generators comprised of bins with latency at 80 ms. Bayesian binning latency detection outperforms the other methods for all dataset sizes. The high, flat error curve of the 2SD method in the 80 Hz peak firing rate condition is due to a consistent underestimation of latency, which is an artifact of Gaussian kernel smoothing combined with a baseline SEM that is small in comparison with the firing rate step at the latency. *Right*: generator with sloping response onsets. We measured the RMS against an assumed latency of 80 ms, even though latency is no longer well defined in these conditions. Our Bayesian binning method is still better than the competitors, despite the fact that a slope is hard to model with bins. Its increase in RMS between 10 trials and 30 trials in the high peak firing condition is due to a flat signal separation maximum (see also Fig. 6, right)

Our method is more accurate than the others in all tested conditions. This is true even if the generator has a sloping response onset (Fig. 7, right) and can no longer be easily modelled by bins. In this case, latency is not as clearly defined as for a step response onset. We took the point of the first rate inflection at 80 ms to be the 'true' latency. Note that this is an additional condition which is not a part of our latency definition. If we had certain knowledge of the generating firing rates, any $S \in (10\text{Hz}, 80\text{Hz})$ would be suitable as a separation level. A consequence of choosing the first point of inflection as 'true' latency is an increase in RMS of Bayesian binning between 10 and 30 trials for the 80 Hz peak, sloping onset condition. This is due to a very flat signal separation maximum (see also Fig. 6, right), i.e. there are many values of S which allow for an almost equally certain separation between ‘firing rate above S’ and ‘firing rate below S’. Since we search for a single maximum, this maximum’s location will then mostly be determined by noise, and not by differences in signal quality. If we wanted to bring the L closer to the first rate inflection point, we would have to optimise a compromise between large $P(L$ exists) and small S. This could be accomplished by adding a weak prior over S which prefers small S. However, this is no longer a ‘minimal’ definition of latency, so we will continue to use our original definition.

4.4 Trial-by-trial latency and firing rate estimation

So far, we computed the model posteriors and all quantities derived thereof on the assumption that there is a single ‘correct’ PSTH from which the data were generated. In other words, we presupposed that the experimentally controlled parameters (e.g. stimulus identity and presentation time) were enough to specify the spike train generating process up to a random element, which is fully modelled by the firing probability. One might object to this model. It is certainly conceivable that for instance latencies and firing rates of the generator vary between trials. Therefore, it would be desirable to be able to compute the posterior distributions of these parameters on a trial-by-trial basis. It is possible to do that with our method, as indicated by the single trials performances in simulations (see Figs. 3 and 7). Figure 8, left, shows a trial-by-trial latency posterior distribution marginalised across all trials to stimuli of high (100%), medium (50%) and low (12.5%) contrast. The high contrast latency posterior was calculated on the same data as those used in Fig. 5, right. While the posterior uncertainty is increased due to the trial-by-trial evaluation, the bulk of the probability is in the same post-stimulus time range (≈75 ms–110 ms, with $S \approx 47$ Hz) as before. Moreover, it is apparent that latency increases with decreasing stimulus contrast, which was also observed in Oram et al. (2002) using a statistical approach (Oram and Perrett 1992; Oram and Perret 1996).

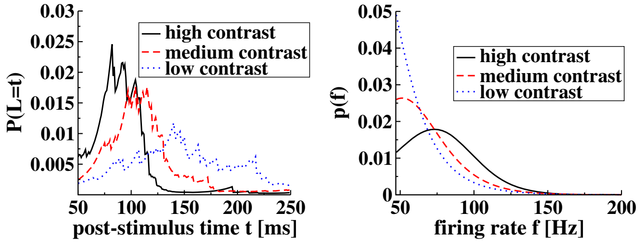

Fig. 8 Trial-by-trial latency and firing rate posteriors for three stimulus contrasts. *Left*: a latency posterior was computed for each trial and then marginalised across all trials for a given contrast. The high contrast posterior was calculated on the same data as the latency posterior in Fig. 5, right. While the posterior uncertainty is increased due to the trial-by-trial evaluation, the bulk of the probability is in the same post-stimulus time range (≈75 ms–110 ms) as before. Reducing stimulus contrast clearly increases latencies. *Right*: firing rate posterior densities in the first bin after the latency

To calculate the posterior distribution of firing rates across trials, one can proceed in a fashion similar to that used for latency: define the probability density that the firing rate at t, $f(t)$, is $\tilde{f}$ given the model parameters as

$$p(f(t)=\tilde{f}|\{k_m\},\{f_m\},M) = \begin{cases} \delta(f_j-\tilde{f}) \text{ if } t \in \{k_{j-1}+1,\ldots,k_j\} \\ 0 \text{ otherwise} \end{cases} \tag{23}$$

where $\delta(x)$ is the Dirac delta function. In words, this probability density is concentrated at the firing rate f_j of that bin which contains the time index t if $f_j=\tilde{f}$. By adding the condition that the lower bound of bin j is equal to the latency, we can compute the probability density of the firing rate in the first bin after the latency, i.e. in the strong transient part of the response. If $\{k_m\}$ and $\{f_m\}$ are given, then this firing rate f_j depends on the latency only through the signal level S, because $f_j \geq S$ (see Eq. (21)). Thus, we can compute the joint probability (density) of ‘latency $L=t$’ and ‘firing rate is $\tilde{f}$’ by multiplying the r.h.s of Eq. (23) with the r.h.s of Eq. (21) if $\tilde{f} \geq S$:

$$p(f(t)=\tilde{f}, L=t|\{k_m\},\{f_m\},M,S) = \begin{cases} p(f(t)=\tilde{f}|\{k_m\},\{f_m\},M) \\ \quad \times P(L=t|\{k_m\},\{f_m\},M,S) \text{ if } \tilde{f} \geq S \\ 0 \;\; \text{otherwise.} \end{cases} \tag{24}$$

Averaging this probability density over the posterior Eq. (8) is done by an algorithm similar to the one used for latency, as detailed in Appendix C. This yields $P(f(t)=\tilde{f}, L=t, \{\vec{z}^i\}|M,S)$. Therefore we have

$$p(f(t)=\tilde{f}|L=t,\{\vec{z}^i\},M,S) = \frac{p(f(t)=\tilde{f}, L=t, \{\vec{z}^i\}|M,S)}{P(L=t,\{\vec{z}^i\}|M,S)} \tag{25}$$

i.e. the probability density of the firing rate being $\tilde{f}$ given that the latency is at t, the signal level is S and the data $\{\vec{z}^i\}$ for a model with M bins. Averaging this firing rate density across trials yields Fig. 8, right. Here, firing rates were found to decrease with stimulus contrast. Furthermore, the posteriors are unimodal—this indicates that modelling the trial-by-trial variations in firing rate by e.g. a mixture of binomial with a unimodal mixing distribution might be a viable strategy.

5 Information-theoretic analysis of latency and firing rate

It is often interesting to quantify the amount of information which a neural response carries about various stimulus parameters. Information theory (Shannon 1948) provides the mathematical framework to address this question: *mutual information* $I(U;C)$ (Cover and Thomas 1991) measures how much we can expect to learn about a (discrete) stimulus parameter C from a (discrete) neural response measure U, and vice versa. Given a joint probability distribution $P(U,C)$, $I(U;C)$ is defined as

$$I(U;C) = \sum_C \sum_U P(U,C) \log\left(\frac{P(U,C)}{P(U)P(C)}\right) \tag{26}$$

where $P(U) = \sum_C P(U,C)$ and likewise for $P(C)$. If a second neural response measure V and the joint probability distribution $P(U,V,C)$ is available, it is possible to define *conditional mutual information* $I(U;C|V)$ and *joint mutual information* $I(U,V;C)$ (Cover and Thomas 1991):

$$I(U;C|V) = \sum_C \sum_U \sum_V P(U,V,C) \times \log\left(\frac{P(U,C|V)}{P(U|V)P(C|V)}\right) \tag{27}$$

$$\begin{aligned} I(U,V;C) &= \sum_C \sum_U \sum_V P(U,V,C) \times \log\left(\frac{P(U,V,C)}{P(U,V)P(C)}\right) \\ &= I(U;C|V) + I(V;C) \end{aligned} \tag{28}$$

$I(U;C|V)$ can be understood as the amount of information we expect to gain about C by observing U if we knew V, whereas $I(U,V;C)$ is the expected information gain about C if we learned the values of both U and V. Extending these definitions to continuous variables is straightforward (Cover and Thomas 1991).

In Sections 4.1 and 4.4, we developed the formalism to compute the posterior distribution of the latency L (Eq. (22)) and the posterior density of the firing rate $f(t)$ in the first bin after latency (Eq. (25)), providing the joint density

$$\begin{aligned} &p(f(t)=\tilde{f}, L=t|\{\vec{z}^i\}, M, S) \\ &\quad = p(f(t)=\tilde{f}|L=t, \{\vec{z}^i\}, M, S)\; P(L=t|\{\vec{z}^i\}, M, S) \end{aligned} \tag{29}$$

which we need to compute joint, conditional and marginal mutual informations between L, $f(t)$ and any stimulus parameter. Note that these distributions/densities are conditioned on the signal level S. So far, we described a procedure to determine S for a single stimulus condition C (see end of Section 4.1). We define S for multi-valued C based on two assumptions:

1. the signal level S is a property of the cell, not of the stimulus. In other words, there is a single S per cell across all C. If S was allowed to vary with C, the choice of S would inject stimulus-related information into the information estimates which is not present in the data.
2. S is determined by maximising the marginal probability of latency existence $P(L \text{ exists}|S)$ (and therefore, signal existence)

$$P(L \text{ exists}|S) = \sum_C P(L \text{ exists}|S,C)P(C) \tag{30}$$

   where $P(C)$ is the prior probability of each stimulus condition, which is controlled by the experimenter.
3. We assume that there is no a-priori dependency between S and C.

Assumption 2 is a consequence of the experimental design which we are about to analyse. Cells and stimuli were selected such that there was at least one stimulus which evoked a strong response, and at least one that evoked a weak response (possible none). Maximising the marginal probability of latency existence thus has the effect of choosing an S such that as many stimulus conditions as possible have a detectable latency. If there is a strong and a weak (but still detectable) response, this procedure chooses a relatively small S such that $P(L \text{ exists}|S,C)$ is high for both C. However, if there is a strong and a non-detectable response, the value of S will be higher, since it will be driven only by the strong response. It remains to be seen if this procedure needs to be adapted for different cell/stimulus choices.

5.1 Results on simulated data

We mentioned in Section 4.2 that the latency posterior inevitably contains information about the change in firing rate at the latency. To illustrate this point, we performed an information-theoretic analysis of a two-stimulus scenario on simulated data. Each stimulus evoked a 50 ms long transient response, followed by a sustained response (duration 250 ms) with a firing rate between the transient and the 10 Hz baseline. In the 'no difference' condition, the two simulated responses had the same underlying generator. We also varied just the firing rate (transient: 100 Hz vs. 30 Hz), just the latency (80 ms vs. 90 ms) or both firing rate and latency. Each dataset contained 10 trials per stimulus and was analysed trial-by-trial (i.e. one PSTH inferred per trial). The average results from 10 repetitions of the simulations are summarised in Table 2. This table shows the mutual informations between stimulus identity C, and the variables:

- E: latency exists, i.e. $L \in \{30\text{ ms}, \ldots, 250\text{ ms}\}$.
- L: $L = t$ for $t \in \{30\text{ ms}, \ldots, 250\text{ ms}\}$, see Eq. (22). Additionally, L has a special value indicating that a latency does not exist (i.e. no transition from below the signal threshold S to above S).
- f: firing probability $f(t) = \tilde{f}$ in the first bin after latency for $\tilde{f} \in [S, 1]$, see Eq. (25). f also has a special value indicating that a firing probability in the first bin after latency does not exist.

Note that both L and f determine E: if latency is somewhere in the latency search interval or if the firing rate in the first bin after latency is somewhere above the signal level, then E is true, otherwise E is false. E can also be read as 'firing rate went above the signal level S somewhere in the latency search interval', and might therefore be viewed as a firing rate related variable, rather than a property of latency. This ambiguity highlights the difficulty of separating firing rate and latency related information, which is due to latency being defined by a firing rate based criterion. We choose to interpret E as carrying firing rate information, since latency is concerned with the *timing* of response onset, rather than just the presence or absence of a response. Thus, information about C in L is given by the conditional mutual information $I(L; C|E)$.

The values in the 'no difference' condition in Table 2 represent the overestimation biases of our method in this scenario. Overestimation of mutual information (and the closely related underestimation of entropy) from small datasets is a well-known problem, and many remedies have been devised for it (Optican et al. 1991; Panzeri and Treves 1996; Nemenman et al. 2004; Paninski 2004; Endres and Földiák 2005). However, most of these methods assume a set of datapoints as a starting point, not a set of posterior distributions. Hence, they can not be applied to our analysis unaltered. Further work will be needed to understand how best to provide, within our analysis framework, information estimates whose overestimation is as small as possible.

If there is only a difference in firing rates, then $I(f; C) > I(L; C|E)$ but $I(L; C|E)$ is still significantly greater than in the 'No difference' condition. In other words, even though the simulated cells were designed to have the same latency (80 ms), the latency posterior distributions inferred from a finite sample carry information about the magnitude of the firing rate change – a large response allows for the determination of latency with greater certainty than a small one. Compare this to the 'difference in L' condition: while $I(L; C|E)$ is about as large as before, $I(L; C|E) > I(f; C)$, i.e. our method is able to distinguish between (un)certainty related and variability related latency information via the information in f. Furthermore, in both 'difference in f' conditions, E contains a large fraction of the firing rate information, i.e. knowing whether the signal threshold was crossed is the most informative aspect of f.

In summary, our method yields the results one would expect for each condition: if the stimulus identity C is encoded in f, then $I(f; C)$ is maximal, if changes in C cause changes in L, $I(L; C|E)$ is maximal. If both L and f are influenced by C, then both can be used together to determine C.

Table 2 Mutual information I in [bit] for simulated neurons with a baseline firing rate of 10 Hz, trial-by-trial analysis

Difference in	$I(E; C)$	$I(L; C\|E)$	$I(f; C)$
No difference	0.002 ± 0.001	0.045 ± 0.004	0.008 ± 0.002
f: 100/30 Hz	0.255 ± 0.023	0.079 ± 0.014	0.314 ± 0.023
L: 80/90 ms	0.007 ± 0.002	0.084 ± 0.010	0.016 ± 0.004
f: 100/30 Hz, L: 80/90 ms	0.206 ± 0.026	0.072 ± 0.007	0.265 ± 0.023

C is stimulus identity, there were two stimuli. L is latency, f is firing rate in the first bin after latency and latency existence is E. The latter is the truth value of the proposition 'Latency is somewhere between 30 ms and 250 ms after stimulus onset'. Difference in f means that the peak firing rates were 30 Hz for one stimulus and 100 Hz for the other, duration of peak response 50 ms, latency 80 ms after stimulus onset. In the 'difference in L' condition, both neurons had a peak firing rate of 100 Hz for 50 ms, with a latency of 80 ms for one stimulus and 90 ms for the other. 'No difference' means that both peak firing rates (100 Hz) and latencies (80 ms) were equal. Errors are SEM computed from 10 repetitions of the simulations. For details, see text

5.2 Results on STSa data

It is known that stimulus contrast influences latency of STSa neurons (Oram et al. 2002; van Rossum et al. 2008). We now examine responses to high-contrast

Table 3 Average trial-by-trial mutual informations and standard errors of the mean (SEM) computed from 29 STSa neurons under high-contrast viewing conditions

Mutual information between C and		Average ± SEM [bit]
Signal existence E	$I(E; C)$	0.0594 ± 0.0191
Latency L given E	$I(L; C\|E)$	0.0650 ± 0.0075
Firing rate f	$I(f; C)$	0.0730 ± 0.0205
Firing rate given latency	$I(f; C\|L)$	0.0136 ± 0.0020
Latency given firing rate	$I(L; C\|f)$	0.0649 ± 0.0077
Joint code	$I(f, L; C)$	0.1379 ± 0.0074

Entropy of stimulus identity C is $H(C) \approx 1$ bit for all cells. E, L and f have the same meaning as in Table 2. Firing rate f in the first bin after latency carries slightly more information about stimulus identity C than latency L. For details, see text

presentations to ask whether latency changes convey stimulus identity related information in the absence of contrast change. The results of a trial-by-trial analysis of mutual informations computed from 29 STSa neurons under high-contrast viewing conditions are shown in Table 3. Entropy of stimulus identity C is $H(C) \approx 1$ bit for all cells. Since $I(f; C) > I(L; C|E)$, firing rate f in the first bin after latency carries slightly more information about C than latency L, but the difference is not significant. The joint code of latency and firing rate is almost as informative as the sum of the individual codes, $I(f, L; C) \approx I(f; C) + I(L; C|E)$. This is also indicated by $I(L; C|f) \approx I(L; C|E)$: the stimulus identity information in firing rate which is redundant with latency is almost completely contained in E. In other words, the most informative firing rate feature is whether the firing rate crosses the signal threshold or not. To decode stimulus identity, we should therefore answer questions about latency and firing rate in the following order of importance: *has* the cell fired above S, *when* has it fired above S, *how much* has it fired above S? While these conclusions are certainly conditioned on our small stimulus set (2 stimuli per cell), the values of the mutual informations are small compared to the theoretical maximum of 1 bit. This makes ceiling effects unlikely.

6 Summary

We have extended our exact Bayesian binning method (Endres et al. 2008) for the estimation of PSTHs. Besides treating uncertainty—a real problem with small neurophysiological datasets—in a principled fashion, it also outperforms several competing methods on real neural data. Amongst the competitors, we found that only BARS (DiMatteo et al. 2001) offers comparable predictive performance. However, BARS requires sampling to compute posterior averages, which can potentially take very long or even get stuck, a problem which we observed on data sets containing only a small number of spikes. Bayesian binning allows for the exact evaluation of posterior averages (within numerical roundoff errors) independent of the contents of the data set. It also offers automatic complexity control because the model posterior can be evaluated. While its computational cost is significant, it is still fast enough to be useful: evaluating the predictive probability takes less than 1s on a modern PC,[2] with a small memory footprint (<10 MB for 512 spiketrains). We showed how our approach can be adapted to extract characteristic features of neural responses in a Bayesian way, e.g. response latencies or firing rate distributions. But we are not restricted these features: we can use our method to compute expectations of any function of the PSTH, subject to the condition that the function depends on the PSTH in a bin-wise fashion. A free software implementation is available at the machine learning open source software repository.[3] This implementation contains a short tutorial, computes expected PSTH and posterior standard deviations, separation level and latency posterior. It also allows for the optimisation of the prior hyperparameters. The code for the information theoretic calculations is available from the authors on request, but it requires a cluster computer to run efficiently: the integration over the posterior distribution of the firing rate needs to be done numerically and is time consuming (≈1–2 days per processor per spiketrain for a trial-by-trial analysis).

The latency alignment procedure of Ventura (2004) was developed to quantify trial-by-trial variation of the response latencies, and as such was not intended to determine an absolute latency estimate. However, Ventura (2004) suggested that the minimal latency estimate from the individual trials could be used. We find this yields a poor estimate of absolute latency which tends to get worse with increasing number of trials. We therefore used the half height method from Gawne et al. (1996a) on the aligned spiketrains to improve the absolute latency estimate. This appears to make the estimate largely independent of the number of trials (see Fig. 7). In the majority of cases, this procedure still underestimates the absolute latency, since the trials

[2] 3.2 GHz Intel Xeon™, SuSE Linux 10.1.

[3] http://www.mloss.org, package ‘binsdfc’.

are aligned to the minimal trial latency. To counter this systematic underestimation, we experimented with shifting all aligned trials by the difference between the total pre-alignment and post-alignment means, thereby restoring the original total mean spike time. However, this did not improve results notably. Aligning trials by mean spike time works well on relatively regular spiketrains (such as the gamma order 8 ISI distribution simulations used in Ventura (2004)). In our simulations with short transients and Bernoulli spike generation, it appears not to work. We would therefore conclude that the poor performance of this method is due to poor estimates of the mean spike time of each trial.

Substituting our observation model (Eq. 2) with any other distribution is straightforward, as long as the replacement is also comprised of bins. One might e.g. model each spike train within a bin by a separate Bernoulli process and mix these with a suitable distribution to capture the inter-trial differences. Alternatively, one could use a model similar to that of Shinomoto and Koyama (2007): choose a Gamma process for the inter-spike intervals and model the time-dependent rate with a bin model.

There are a number of other approaches to PSTH/SDF estimation which were not included in our comparisons. Perhaps most noteworthy (from a Bayesian perspective) are Shinomoto and Koyama (2007) and a recent Gaussian process model (Cunningham et al. 2008). We have not yet directly compared our method to either of them. Comparisons to Cunningham et al. (2008) and Shinomoto and Koyama (2007) will be interesting future work, once the authors of these works release their code.

Finally, we used our approach to compute exact (up to roundoff errors) expectations of information-theoretic quantities, e.g. mutual informations between latency, firing rate and stimulus identity. We demonstrated that STSa neurons convey most of the information about stimulus identity through changes in firing rate. Specifically, we found that the crossing of a signal threshold S is the most informative firing rate feature. However, extra information about stimulus identity can be gained by looking at the response latency.

Acknowledgements D. Endres was supported by a Medical Research Council (UK) special training fellowship in bioinformatics G0501319. Both authors would like to thank P. Földiak and J. Schindelin for stimulating discussions and comments on the manuscript. Data collection was supported by EU framework grant (FP5-Mirror) to M. Oram. We thank the unknown reviewers for their clarifying suggestions and references.

Appendix A: Computing the evidence with dynamic programming

The evidence, or marginal likelihood of a model with M bins is given by (see Eq. (9)):

$$P(\{\vec{z}^i\}|M) = \sum_{k_{M-1}=M-1}^{T-2} \sum_{k_{M-2}=M-2}^{k_{M-1}-1} \dots$$
$$\dots \sum_{k_0=0}^{k_1-1} P(\{\vec{z}^i\}|\{k_m\}, M) P(\{k_m\}|M) \qquad (31)$$

where the summation boundaries are chosen such that the bins are non-overlapping and contiguous and

$$P(\{\vec{z}^i\}|\{k_m\}, M)$$
$$= \int_0^1 d\{f_m\} P(\{\vec{z}^i\}|\{f_m\}, \{k_m\}, M) p(\{f_m\}|M). \qquad (32)$$

Recall that the probability of a (multi)set of spiketrains $\{\vec{z}^i\} = \{z_1, \dots, z_N\}$, assuming independent generation, is given by Eq. (3):

$$P(\{\vec{z}^i\}|\{f_m\}, \{k_m\}, M) = \prod_{i=1}^{N} \prod_{m=0}^{M} f_m^{s(\vec{z}^i,m)} (1-f_m)^{g(\vec{z}^i,m)}$$
$$= \prod_{m=0}^{M} f_m^{s(\{\vec{z}^i\},m)} (1-f_m)^{g(\{\vec{z}^i\},m)} \qquad (33)$$

where $s(\{\vec{z}^i\}, m) = \sum_{i=1}^{N} s(\vec{z}^i, m)$ is the number of spikes in all spiketrains in bin m and $g(\{\vec{z}^i\}, m) = \sum_{i=1}^{N} g(\vec{z}^i, m)$ is the number of all non-spikes, or gaps. The prior of the firing rates (Eq. (5)) is

$$p(\{f_m\}|M) = \prod_{m=0}^{M} \mathrm{B}(f_m; \sigma_m, \gamma_m). \qquad (34)$$

The integrals in Eq. (32) can be evaluated by virtue of Eqs. (33 and 34):

$$P(\{\vec{z}^i\}|\{k_m\}, M) = \prod_{m=0}^{M} \frac{\mathrm{B}(s(\{\vec{z}^i\}, m)+\sigma_m, g(\{\vec{z}^i\}, m)+\gamma_m)}{\mathrm{B}(\sigma_m, \gamma_m)} \qquad (35)$$

where $\mathrm{B}(x, y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}$ is Euler's Beta function (Davis 1972). Equation (35) is a product with one factor per bin, and each factor depends only on spike/gap counts and prior parameters in that bin. To compute Eq. (31), we can therefore use an approach very similar to that described in (Endres and Földiák 2005; Endres 2006) in the context of density estimation and in (Hutter 2006, 2007) for Bayesian function approximation: define the function

$$\mathrm{getIEC}(k_s, k_e, m) := \mathrm{B}(s(\{\vec{z}^i\}, k_s, k_e) + \sigma_m, g(\{\vec{z}^i\}, k_s, k_e) + \gamma_m) \qquad (36)$$

where $s(\{\vec{z}^i\}, k_s, k_e)$ is the number of spikes and $g(\{\vec{z}^i\}, k_s, k_e)$ is the number of gaps in $\{\vec{z}^i\}$ between the start interval k_s and the end interval k_e (both included). Furthermore, collect all contributions to Eq. (31) that do not depend on the data (i.e. $\{\vec{z}^i\}$) and store them in the array $\mathrm{pr}[M]$:

$$\mathrm{pr}[M] := \frac{\prod_{m=0}^{M} \frac{1}{\mathrm{B}(\sigma_m, \gamma_m)}}{\binom{T-1}{M}}. \qquad (37)$$

Substituting Eq. (35) into Eq. (31) and using the definitions (36) and (37), we obtain

$$P(\{\vec{z}^i\}|M) \propto \sum_{k_{M-1}=M-1}^{T-2} \dots \sum_{k_0=0}^{k_1-1} \prod_{m=1}^{M} \mathrm{getIEC}(k_{m-1}+1, k_m, m) \times \mathrm{getIEC}(0, k_0, 0) \qquad (38)$$

with $k_M = T - 1$ and the constant of proportionality being $\mathrm{pr}[M]$. Since the factors on the r.h.s. depend only on two consecutive bin boundaries each, it is possible to apply dynamic programming (Bertsekas 2000): rewrite the r.h.s. by 'pushing' the sums as far to the right as possible:

$$P(\{\vec{z}^i\}|M) \propto \sum_{k_{M-1}=M-1}^{T-2} \mathrm{getIEC}(k_{M-1}+1, T-1, M) \times \sum_{k_{M-2}=M-2}^{k_{M-1}-1} \mathrm{getIEC}(k_{M-2}+1, k_{M-1}, M-1) \times \dots \sum_{k_0=0}^{k_1-1} \mathrm{getIEC}(k_0+1, k_1, 1)\mathrm{getIEC}(0, k_0, 0). \qquad (39)$$

Evaluating the sum over k_0 requires $O(T)$ operations (assuming that $T \gg M$, which is likely to be the case in real-world applications). As the summands depend also on k_1, we need to repeat this evaluation $O(T)$ times, i.e. summing out k_0 for all possible values of k_1 requires $O(T^2)$ operations. This procedure is then repeated for the remaining $M-1$ sums, yielding a total computational effort of $O(MT^2)$. Thus, initialise the array $\mathrm{subE}_0[k] := \mathrm{getIEC}(0, k, 0)$, and iterate for all $m = 1, \dots, M$:

$$\mathrm{subE}_m[k] := \sum_{r=m-1}^{k-1} \mathrm{getIEC}(r+1, k, m)\mathrm{subE}_{m-1}[r], \qquad (40)$$

Table 4 Computing the evidences of models with up to M bin boundaries

```
for $k := 0 \dots T-1$ : subE[$k$] := getIEC(0, $k$, 0)
E[0] := subE[$T-1$] $\times$ pr[0]
for $m := 1 \dots M$ :
    (a) if $m = M$ then $l := T-1$ else $l := m$
    (b) for $k := T-1 \dots l$
            subE[$k$] := $\sum_{r:=m-1}^{k-1}$ subE[$r$] $\times$ getIEC($r+1, k, m$)
    (c) E[$m$] = subE[$T-1$] $\times$ pr[$m$]
return E[]
```

A close look at Eq. (39) reveals that while we sum over k_{M-1}, we need $\mathrm{subE}_{M-1}[k]$ for $k = M-1; \dots; T-2$ to compute the evidence of a model with its latest boundary at $T-1$. We can, however, compute $\mathrm{subE}_{M-1}[T-1]$ with little extra effort, which is, up to a factor $\mathrm{pr}[M-1]$, equal to $P(\{\vec{z}^i\}|M-1)$, i.e. the evidence for a model with $M-1$ bin boundaries. Moreover, having computed $\mathrm{subE}_m[k]$, we do not need $\mathrm{subE}_{m-1}[k-1]$ anymore. Hence, the array $\mathrm{subE}_{m-1}[k]$ can be reused to store $\mathrm{subE}_m[k]$, if overwritten in reverse order. Table 4 shows this algorithm in pseudo-code ($\mathrm{E}[m]$ contains the evidence of a model with m bin boundaries inside $[t_{\min}, t_{\max}]$ after termination).

Appendix B: Computing the posterior distribution of the latency

We compute the joint probability of the latency $L = t$ and the observed spiketrains $\{\vec{z}^i\}$ given the number of bins and the signal separation level S via

$$P(L=t, \{\vec{z}^i\}|M, S) = \sum_{k_{M-1}=M-1}^{T-2} \sum_{k_{M-2}=M-2}^{k_{M-1}-1} \dots \sum_{k_0=0}^{k_1-1} P(L=t, \{\vec{z}^i\}, \{k_m\}|M, S) \qquad (41)$$

where

$$\begin{aligned}P(L=t,\{\vec{z}^i\},\{k_m\}|M,S)&\\ =\int_0^1 d\{f_m\}\,&P(L=t|\{k_m\},\{f_m\},M,S)\\ &\times\ p(\{\vec{z}^i\},\{f_m\},\{k_m\}|M).\end{aligned}\tag{42}$$

Note that $P(L=t|\{k_m\},\{f_m\},M,S)$ is the r.h.s of Eq. (21) and $p(\{\vec{z}^i\},\{f_m\},\{k_m\}|M)$ is the numerator of the r.h.s. of Eq. (8). As a consequence of Eq. (21), the only nonzero contributions to the average are models which have a (lower) bin boundary at t. Assume t was at the lower bound of bin j, i.e. at $t=k_{j-1}+1$ (the $\{k_m\}$ are inclusive upper bin boundaries, as defined above). Carrying out the integrals over the $\{f_m\}$ yields:

$$\begin{aligned}P(L=t,\{\vec{z}^i\},\{k_m\}|M,S)&\\ =\int_0^1 d\{f_m\}\,&P(L=t|\{k_m\},\{f_m\},M,S)\\ &\times\ p(\{\vec{z}^i\},\{f_m\},\{k_m\}|M)\\ =\int_0^S df_0\ldots&\int_0^S df_{j-1}\int_S^1 df_j\int_0^1 df_{j+1}\ldots\\ \ldots&\int_0^1 df_M\,p(\{\vec{z}^i\},\{f_m\},\{k_m\}|M)\end{aligned}\tag{43}$$

The integration boundaries in the last line of Eq. (43) are a consequence of our latency definition: all bins $m<j$ will contribute to the integral only as long as $f_m<S$, hence the upper bound of their integrals is at S. Bin j contributes only if $f_j\geq S$, thus the lower bound of the integral over f_j is S. The bins $m>j$ are not affected by the latency probability (Eq. (21)) whence their integrals still run from 0 to 1. By virtue of Eqs. (3) and (5), we obtain

$$\begin{aligned}P(L=t,\{\vec{z}^i\},\{k_m\}|M,S)&\\ =\prod_{m=0}^{j-1}&B_S(s(\{\vec{z}^i\},m)+\sigma_m,g(\{\vec{z}^i\},m)+\gamma_m)\\ &\times\ \tilde{B}_S(s(\{\vec{z}^i\},j)+\sigma_m,g(\{\vec{z}^i\},j)+\gamma_m))\\ &\times\prod_{m=j+1}^{M}B(s(\{\vec{z}^i\},m)+\sigma_m,g(\{\vec{z}^i\},m)+\gamma_m)\\ &\times\prod_{m=0}^{M}\frac{1}{B(\sigma_m,\gamma_m)}P(\{k_m\}|M)\end{aligned}\tag{44}$$

where $B_S(a,b)=\int_0^S t^{a-1}(1-t)^{b-1}dt$ is the incomplete Beta function (Davis 1972) and $\tilde{B}_S(a,b)=\int_S^1 t^{a-1}(1-t)^{b-1}dt=B(a,b)-B_S(a,b)$ is its complement. Note that up to the factor $P(\{k_m\}|M)$, Eq. (44) is basically Eq. (35) with some of the Beta functions replaced by incomplete Beta functions. Hence, the remaining summations over the $\{k_m\}$ can be carried out by using

$$\text{getIEC}_L(k_s,k_e,m):=\\ =\begin{cases}B_S(s(\{\vec{z}^i\},k_s,k_e)\\ \quad+\sigma_m,g(\{\vec{z}^i\},k_s,k_e)+\gamma_m) & \text{if } k_e<t\\ \tilde{B}_S(s(\{\vec{z}^i\},k_s,k_e)\\ \quad+\sigma_m,g(\{\vec{z}^i\},k_s,k_e)+\gamma_m)) & \text{if } k_s=t\\ B(s(\{\vec{z}^i\},k_s,k_e)\\ \quad+\sigma_m,g(\{\vec{z}^i\},k_s,k_e)+\gamma_m) & \text{if } k_s>t\\ 0 & \text{otherwise}\end{cases}\tag{45}$$

instead of getIEC(k_s,k_e,m) (Eq. (36)) in the evidence computation algorithm. This procedure yields the desired $P(L=t,\{\vec{z}^i\}|M,S)$. The above derivation is an instance of the general framework for computing expectations of functions of bin boundaries and firing probabilities described in Endres and Földiák (2005).

Appendix C: Computing the posterior density of the firing rate

We compute the joint probability density of the firing rate $f(t)=\tilde{f}$, the latency $L=t$ and the observed spiketrains $\{\vec{z}^i\}$ given the number of bins and the signal separation level S via

$$\begin{aligned}P(f(t)=\tilde{f},L=t,\{\vec{z}^i\}|M,S)=&\\ =\sum_{k_{M-1}=M-1}^{T-2}\ \sum_{k_{M-2}=M-2}^{k_{M-1}-1}&\ldots\\ \ldots\sum_{k_0=0}^{k_1-1}&P(f(t)=\tilde{f},L=t,\{\vec{z}^i\},\{k_m\}|M,S)\end{aligned}\tag{46}$$

where

$$\begin{aligned}p(f(t)=\tilde{f},L=t,\{\vec{z}^i\},\{k_m\}|M,S)&\\ =\int_0^1 d\{f_m\}\,&P(f(t)=\tilde{f},L=t|\{k_m\},\{f_m\},M,S)\\ &\times\ p(\{\vec{z}^i\},\{f_m\},\{k_m\}|M)\end{aligned}\tag{47}$$

Note that $P(f(t) = \tilde{f}, L = t | \{k_m\}, \{f_m\}, M, S)$ is the r.h.s of Eq. (24) and $p(\{\vec{z}^i\}, \{f_m\}, \{k_m\} | M)$ is the numerator of the r.h.s. of Eq. (8). As a consequence of Eq. (24), the only nonzero contributions to the average are models which have a (lower) bin boundary at t and $f(t) \geq S$. Assume t was at the lower bound of bin j, i.e. at $t = k_{j-1} + 1$ (the $\{k_m\}$ are inclusive upper bin boundaries, as defined above). Integrating out the $\{f_m\}$ yields

$$\begin{aligned}
&p(f(t) = \tilde{f}, L = t, \{\vec{z}^i\}, \{k_m\} | M, S) \\
&\quad = \prod_{m=0}^{j-1} B_S(s(\{\vec{z}^i\}, m) + \sigma_m, g(\{\vec{z}^i\}, m) + \gamma_m) \\
&\qquad \times \tilde{f}^{s(\{\vec{z}^i\}, j) + \sigma_m - 1} (1 - \tilde{f})^{g(\{\vec{z}^i\}, j) + \gamma_m) - 1} \\
&\qquad \times \prod_{m=j+1}^{M} B(s(\{\vec{z}^i\}, m) + \sigma_m, g(\{\vec{z}^i\}, m) + \gamma_m) \\
&\qquad \times \prod_{m=0}^{M} \frac{1}{B(\sigma_m, \gamma_m)} P(\{k_m\} | M) \qquad (48)
\end{aligned}$$

where the second line is a result of Eq. (3) and (5) multiplied with the Dirac delta function in Eq. (23). Hence, the remaining summations over the $\{k_m\}$ can be carried out by using

$$\mathrm{getIEC}_{f,L}(k_s, k_e, m) :=$$

$$\begin{cases}
B_S(s(\{\vec{z}^i\}, k_s, k_e) & \\
\quad + \sigma_m, g(\{\vec{z}^i\}, k_s, k_e) + \gamma_m) & \text{if } k_e < t \\
\tilde{f}^{s(\{\vec{z}^i\}, k_s, k_e) + \sigma_m - 1} & \\
\quad \times (1 - \tilde{f})^{g(\{\vec{z}^i\}, k_s, k_e) + \gamma_m - 1} & \text{if } k_s = t \\
B(s(\{\vec{z}^i\}, k_s, k_e) & \\
\quad + \sigma_m, g(\{\vec{z}^i\}, k_s, k_e) + \gamma_m) & \text{if } k_s > t \\
0 & \text{otherwise}
\end{cases} \qquad (49)$$

instead of $\mathrm{getIEC}(k_s, k_e, m)$ (Eq. (36)) in the evidence computation algorithm. Thus we obtain $P(f(t) = \tilde{f}, L = t, \{\vec{z}^i\} | M, S)$.

References

Alitto, H. J., & Usrey, W. M. (2004). Influence of contrast on orientation and temporal frequency tuning in ferret primary visual cortex. *Journal of Neurophysiology, 91*, 2797–2808.

Barraclough, N., Xiao, D., Baker, C., Oram, M., & Perrett, D. (2005). Integration of visual and auditory information by superior temporal sulcus neurons responsive to the sight of actions. *Journal of Cognitive Neuroscience, 17*, 377–391.

Berenyi, A., Benedek, G., & Nagy, A. (2007). Double sliding-window technique: A new method to calculate the neuronal response onset latency. *Brain Research, 1178*, 141–148.

Berger, J. (1985). *Statistical decision theory and Bayesian analysis*. New York: Springer.

Bertsekas, D. P. (2000). *Dynamic programming and optimal control*. Nashua: Athena.

Cover, T., & Thomas, J. (1991). *Elements of information theory*. New York: Wiley.

Cunningham, J., Yu, B., Shenoy, K., & Sahani, M. (2008). Inferring neural firing rates from spike trains using Gaussian processes. In J. Platt, D. Koller, Y. Singer, & S. Roweis (Eds.), *Advances in neural information processing systems 20*. Cambridge: MIT.

Davis, P. (1972). Gamma function and related functions. In M. Abramowitz & I. Stegun (Eds.), *Handbook of mathematical functions*. New York: Dover.

DiMatteo, I., Genovese, C. R., & Kass, R. E. (2001). Bayesian curve-fitting with free-knot splines. *Biometrika, 88*(4), 1055–1071.

Edwards, R., Xiao, D., Keysers, C., Földiák, P., & Perrett, D. I. (2003). Color sensitivity of cells responsive to complex stimuli in the temporal cortex. *Journal of Neurophysiology, 90*(2), 1245–1256. doi:10.1152/jn.00524.2002, http://jn.physiology.org/cgi/content/abstract/90/2/1245, http://jn.physiology.org/cgi/reprint/90/2/1245.pdf.

Eifuku, S., De Souza, W. C., Tamura, R., Nishijo, H., & Ono, T. (2004). Neuronal correlates of face identification in the monkey anterior temporal cortical areas. *Journal of Neurophysiology, 91*, 358–371.

Endres, D. (2006). *Bayesian and information-theoretic tools for neuroscience*. Ph.D. thesis, School of Psychology, University of St. Andrews, U.K. http://hdl.handle.net/10023/162.

Endres, D., & Földiák, P. (2005). Bayesian bin distribution inference and mutual information. *IEEE Transactions on Information Theory, 51*(11).

Endres, D., Oram M., Schindelin, J., & Földiák, P. (2008). Bayesian binning beats approximate alternatives: Estimating peri-stimulus time histograms. In J. Platt, D. Koller, Y. Singer, & S. Roweis (Eds.), *Advances in neural information processing systems 20*. Cambridge: MIT.

Földiák, P., Xiao, D., Keysers, C., Edwards, R., & Perrett, D. I. (2004). Rapid serial visual presentation for the determination of neural selectivity in area STSa. *Progress in Brain Research, 144*, 107–116.

Friedman, H.S., & Priebe, C.E. (1998). Estimating stimulus response latency. *Journal of Neuroscience Methods, 83*, 185–194.

Fries, P., Neuenschwander, S., Engel, A. K., Goebel, R., & Singer, W. (2001). Rapid feature selective neuronal synchronization through correlated latency shifting. *Nature Neuroscience, 4*, 194–200.

Gabel, S. F., Misslisch, H., Schaafsma, S. J., & Duysens, J. (2002). Temporal properties of optic flow responses in the ventral intraparietal area. *Visual Neuroscience, 19*, 381–388.

Gawne, T. J., Kjaer, T. W., Hertz, J. A., & Richmond, B. J. (1996a). Adjacent visual cortical complex cells share about 20% of their stimulus-related information. *Cerebral Cortex, 6*(3), 482–489.

Gawne, T. J., Kjaer, T. W., & Richmond, B. J. (1996b). Latency: Another potential code for feature binding in striate cortex. *Journal of Neurophysiology, 76*, 1356–1360.

Hanes, D. P., Thompson, K. G., & Schall, J. D. (1995). Relationship of presaccadic activity in frontal eye field and supplementary eye field to saccade initiation in Macaque–Poisson spike train analysis. *Experimental Brain Research, 103*, 85–96.

Heil, P., & Irvine, D. R. F. (1997). First-spike timing of auditory-nerve fibers and comparison with auditory cortex. *Journal of Neurophysiology, 78*, 2438–2454.

Hurley, L. M., & Pollak, G. D. (2005). Serotonin shifts first-spike latencies of inferior colliculus neurons. *Journal of Neuroscience, 25*, 7876–7886.

Hutter, M. (2006). *Bayesian regression of piecewise constant functions*. Tech. Rep. arXiv:math/0606315v1, IDSIA-14-05.

Hutter, M. (2007). Exact Bayesian regression of piecewise constant functions. *Journal of Bayesian Analysis, 2*(4), 635–664.

Kiani, R., Esteky, H., & Tanaka, K. (2005). Differences in onset latency of Macaque inferotemporal neural responses to primate and non-primate faces. *Journal of Neurophysiology, 94*(2), 1587–1596. doi:10.1152/jn.00540.2004, http://jn.physiology.org/cgi/content/abstract/94/2/1587, http://jn.physiology.org/cgi/reprint/94/2/1587.pdf.

Lee, J., Williford, T., & Maunsell, J. H. R. (2007). Spatial attention and the latency of neuronal responses in macaque area V4. *Journal of Neuroscience, 27*, 9632–9637.

Liu, Z., & Richmond, B. J. (2000). Response differences in monkey te and perirhinal cortex: Stimulus association related to reward schedules. *Journal of Neurophysiology, 83*, 1677–1692.

Loader, C. (1997). LOCFIT: An introduction. *Statistical Computing and Graphics Newsletter, 8*(1), 11–17. http://www.stat-computing.org/newsletter.

Loader, C. (1999). *Local regression and likelihood*. New York: Springer.

Luczak, A., Bartho, P., Marguet, S. L., Buzsaki, G., & Harris, K. D. (2007). Sequential structure of neocortical spontaneous activity *in vivo*. *Proceedings of the National Academy of Sciences of the United States of America, 104*, 347–352.

Maunsell, J. H., & Gibson, J. R. (1992). Visual response latencies in striate cortex of the macaque monkey. *Journal of Neurophysiology, 68*, 1332–1344.

Nawrot, M. P., Aertsen, A., & Rotter, S. (2003). Elimination of response latency variability in neuronal spike trains. *Biological Cybernetics, 88*, 321–334.

Nemenman, I., Bialek, W., & van Steveninck, R. (2004). Entropy and information in neural spike trains: Progress on the sampling problem. *Physical Review E, 69*(5), 056111. http://link.aps.org/abstract/PRE/v69/e056111.

Nowak, L. G., Munk, M. H. J., Girard, P., & Bullier, J. (1995). Visual latencies in areas v1 and v2 of the Macaque monkey. *Visual Neuroscience, 12*, 371–384.

Optican, L., Gawne, T., Richmond, B., & Joseph, P. (1991). Unbiased measures of transmitted information and channel capacity from multivariate neuronal data. *Biological Cybernetics, 65*, 305–310.

Oram, M., & Perret, D. (1996). Integration of form and motion in the anterior superior temporal polysensory area (stpa). of the Macaque monkey. *Journal of Neurophysiology, 19*, 109–129.

Oram, M. W., & Perrett, D. I. (1992). Time course of neural responses discriminating different views of the face and head. *Journal of Neurophysiology, 68*(1), 70–84.

Oram, M. W., Xiao, D., Dritschel, B., & Payne, K. (2002). The temporal precision of neural signals: A unique role for response latency? *Philosophical Transactions of the Royal Society, Series B, 357*, 987–1001.

Paninski, L. (2004). Estimating entropy on m bins given fewer than m samples. *IEEE Transactions on Information Theory, 50*(9), 2200–2203.

Panzeri, S., & Treves, A. (1996). Analytical estimates of limited sampling biases in different information measures. *Network: Computation in Neural Systems, 7*, 87–107.

Perrett, D. I., Oram, M. W., Harries, M. H., Bevan, R., Hietanen, J. K., & Benson P. J. (1991). Viewer-centered and object-centered coding of heads in the Macaque temporal cortex. *Experimental Brain Research, 86*, 159–173.

Press, W., Flannery, B., Teukolsky, S., & Vetterling, W. (1986). *Numerical recipes in C: The art of scientific computing*. New York: Cambridge University Press.

Richmond, B. J., & Optican, L. M. (1987a). Temporal encoding of two-dimensional patterns by single units in primate inferior temporal cortex. II. Quantification of response waveform. *Journal of Neurophysiology, 57*(1), 147–161.

Richmond, B. J., & Optican, L. M. (1987b). Temporal encoding of two-dimensional patterns by single units in primate inferior temporal cortex. III. Information theoretic analysis. *Journal of Neurophysiology, 57*(1), 162–178.

Richmond, B. J., Oram, M. W., & Wiener, M. C. (1999). Response features determining spike times. *Neural Plasticity, 6*, 133–145.

Sary, G., Koteles, K., Chadaide, Z., Tompa, T., & Benedek, G. (2006). Task-related modulation in the monkey inferotemporal cortex. *Brain Research, 1121*, 76–82.

Schmolesky, M. T., Wang, Y., Hanes, D. P., Thompson, K. G., Leutgeb, S., Schall, J. D., et al. (2006). Signal timing across the macaque visual system. *Journal of Neurophysiology, 79*, 3272–3278.

Shannon, C. E. (1948). The mathematical theory of communication. *The Bell Systems Technical Journal, 27*, 379–423, 623–656.

Shimazaki, H., & Shinomoto, S. (2007a). Kernel width optimization in the spike-rate estimation. In R. Budelli, A. Caputi, & L. Gomez (Eds.), *Neural coding 2007* (pp. 143–146). http://neuralcoding2007.edu.uy.

Shimazaki, H., & Shinomoto, S. (2007b). A method for selecting the bin size of a time histogram. *Neural Computation, 19*(6), 1503–1527.

Shimazaki, H., & Shinomoto, S. (2007c). A recipe for optimizing a time-histogram. In B. Schölkopf, J. Platt, & T. Hoffman (Eds.), *Advances in neural information processing systems* (Vol. 19, pp. 1289–1296). Cambridge: MIT.

Shinomoto, S., & Koyama, S. (2007). A solution to the controversy between rate and temporal coding. *Statistics in Medicine, 26*, 4032–4038.

Stecker G. C., & Middlebrooks J. C. (2003). Distributed coding of sound locations in the auditory cortex. *Biological Cybernetics 89*, 341–349

Sugase-Miyamoto, Y., & Richmond, B. J. (2005). Neuronal signals in the monkey basolateral amygdala during reward schedules. *Journal of Neuroscience, 25*, 11071–11083.

Syka, J., Popelar, J., Kvasnak, E., & Astl, J. (2000). Response properties of neurons in the central nucleus and external send dorsal cortices of the inferior colliculus in guinea pig. *Experimental Brain Research, 133*, 254–266.

Tamura, H., & Tanaka, K. (2001). Visual response properties of cells in the ventral and dorsal parts of the macaque inferotemporal cortex. *Cerebral Cortex, 11*, 384–399.

Tanaka, M., & Lisberger, S. G. (2002). Role of arcuate frontal cortex of monkeys in smooth pursuit eye movements. I. Basic response properties to retinal image motion and position. *Journal of Neurophysiology, 87*, 2684–2699.

Thompson, K. G., Hanes, D. P., Bichot, N. P., & Schall, J. D. (1996). Perceptual and motor processing stages identified in the activity of Macaque frontal eye field neurons during visual search. *Journal of Neurophysiology, 76*, 4040–4055.

Tovee, M. J., Rolls, E. T., Treves, A., & Bellis, R. P. (1993). Information encoding and the responses of single neurons in the primate temporal visual cortex. *Journal of Neurophysiology, 70*(2). 640–654. http://jn.physiology.org/cgi/content/abstract/70/2/640, http://jn.physiology.org/cgi/reprint/70/2/640.pdf.

van Rossum, M. C. W., van der Meer, M. A. A., Xiao, D., & Oram, M. W. (2008). Adaptive integration in the visual cortex by depressing recurrent cortical circuits. *Neural Computation, 20*(7), 1847–1872. http://neco.mitpress.org/cgi/content/abstract/20/7/1847, http://neco.mitpress.org/cgi/reprint/20/7/1847.pdf.

Ventura, V. (2004). Testing for and estimating latency effects for Poisson and non-Poisson spike trains. *Neural Computation, 16*(11), 2323–2349. doi:10.1162/0899766041941952.