Statistical Smoothing of Neuronal Data

Robert E. Kass, Valérie Ventura, and Can Cai

kass@stat.cmu.edu

Department of Statistics

and the Center for the Neural Basis of Cognition Carnegie Mellon University

Grateful acknowledgments to Chris Baker, Emery Brown,

Carl Olson, Julie Rollenhagen, and Heidi Sestrich.



The raster and PSTH plot is a wonderful graphic. The raster displays the complete set of spike times for all trials on a single neuron in a particular exponential condition, while the PSTH accumlates these to show the way the firing rate varies accross time. One reason the PSTH works well is that our eye is able to smooth the PSTH so that we see the temporal evolution of the firing rate.



Once we articulate the goal of estimating the firing rate, however, it is possible to improve the PSTH by smoothing (often called filtering), as shown in the figure. This talk will describe physiological motivation for smoothing and methods for carrying it out. We will focus, especially, on a method called BARS that produces nice, smooth curves like the one shown here.



At the outset we would like to be clear that when we speak of "estimating the firing rate" we mean that we will use the data to produce an estimate of the *instantaneous* firing rate, which we write as $\lambda(t)$, at each time t, where t varies across the whole range of experimentally interesting values. In other words, we are interested in estimating the *curve* described by $-\lambda(t)$.



In statistics, we usually write an estimate obtained from data with a hat, so $\lambda(t)$ should really be written as $\hat{\lambda}(t)$. This emphasizes the distinction between an unknown "true" curve and an estimate of it.



Why do we care about instantaneous firing rate? Sometimes, questions of interest require it. For example, a study in the lab of Carl Olson, our colleague at the Center for the Neural Basis of Cognition in Pittsburgh, examined neurons in the Supplementary Eye Field (SEF) when a monkey moved his eyes in response to either an explicit external cue (the point to



which the eyes were to move was illuminated) or an internally-generated translation of a complex cue (a particular pattern at fixation point determined the location to which the monkey was to move his eyes). We were interested in the time at which maximal firing rate was achieved, and the delay of this maximum for the internally-generated cue compared to the external cue.



It would be possible to use the PSTH to estimate the time at which the maximal firing rate is achieved: find the highest peak ...



then the time at which it occurs. However, as can be seen in the figure, that estimate would be noisy.



A more accurate method is to first fit a smooth curve ...



and repeat the process for the curve....



This produces a different value, which better represents the time of peak firing rate under the rather natural assumption that the firing rate is varying smoothly.



A second example again comes from Carl Olson's lab, this time involving neurons from inferotemporal (IT) cortex. In one condition a stimulus was displayed alone, while in the second condition a pair of distractors were also illuminated. For the second condition the damped oscillatory response was more pronounced. One way to quantify this is to measure the peak-to-trough differences. Again, these may be computed from the fitted curves.



It is also worth pointing out that, as this figure shows, it can be easier to compare two curves than to compare two histograms. Here, the two curves are overlayed on top of each other. Using the curves we have great flexibility in quantifying the distinctions we perceive. The statistical methods we discuss can be applied to essentially any feature we wish to analyze. Probability density for set of spikes s_1, \ldots, s_n

$$p(s_1,\ldots,s_n) = e^{-\int_0^T \lambda(t)dt} \prod_{k=1}^n \lambda(s_k)$$

(in non-Poisson case $\lambda(t)$ is generalized; we return to this below)

A second reason that we care about instantaneous firing rate is that probability models make efficient use of the data, and the function $\lambda(t)$ is needed in writing down a probability model, as shown here. This kind of model may be fitted using either maximum likelihood or Bayesian methods, which often provide nearly identical fits, both methods being optimal in the sense of statistical efficiency. The issue of efficiency is particularly important when studying trial-to-trial variation (and its relation to effects such as response time) or when making real-time predictions (decoding) from multiple neurons recorded simultaneously.



To illustrate the statistical efficiency of smoothing, consider data simulated from the true firing rate function shown in this figure.



When we simulate 16 trials (the number of trials in the SEF data shown earlier) we get the PSTH shown here.



We then use the spline-based smooth fit, shown next as "estimated rate."



We repeated this three more times to show that there is some variability in the fits: each of the 4 times we generate new data, we get a new PSTH and a new smooth fitted curve. Now consider what happens when we repeat this process 1000 times, each time using the same firing rate function. We can evaluate how close each of the 1000 PSTH and spline-based fits are to the true firing rate curve and, thus, we can compare the accuracy of the two methods.



This figure shows the 95% bands for the PSTH: 95% of the 1000 PSTH values (at each point in time t) fall inside the wide pair of bands surrounding the true firing rate curve. In this figure each of the 1000 simulated data sets was based on 16 trials.



We summarize numerically by computing the mean integrated squared error (MISE), a measure of the amount by which the PSTH misses the true value, on average. It is 4.68 (spikes/sec)².



We next do the same for the spline-fitted smooth curves: we find for this method that 95% of the fitted values fall inside the much narrower pair of bands (shown as small dotted bands).



Furthermore, the mean integrated squared error is now only 0.34, which is 14 times smaller than that for the PSTH. This means that when the PSTH is used to estimate firing rate, 14 times as much data are needed.



In this last version of the figure we now add a pair of bands obtained from the PSTH when 224 trials are used rather than 16 (because $224 = 14 \times 16$). This shows that the accuracy in estimating the firing rate using spline smoothing based on 16 trials is the same as the accuracy using the PSTH based on 224 trials. Clearly it is very much better to use smoothing when estimating the instantaneous firing rate.



There are many ways to accomplish the smoothing or filtering. We happen to like spline-based methods, which we will explain briefly in a moment. In this figure we return to the SEF data and show a PSTH together with a spline-based smooth version of it.



In this overlay we have added an alternative smoothed version of the PSTH, this time produced using a Gaussian filter, or, what in the statistical literature is called a kernel density estimate. The two methods give very similar fitted curves. The reason the two methods agree well is that the firing rate is in this case varying quite slowly.



A different example, now from another experiment on IT neurons from Carl Olson's lab, shows a case in which the Gaussian filter has difficulty. Note the wiggliness of the fitted curve it produces, together with its inability to track the quick increase in firing rate. The difficulty here is due to the uneven variation of the function: it begins relatively flat, then at around 100 ms after time 0 it has a very steep increase, and then has a relatively smooth decline. The Gaussian filter, which is linear, has a fixed amount of smoothing it applies throughout the range of time. What is needed, instead, is a variable (nonlinear) amount of smoothing, that is, what we might call an *adaptive* method.



An adaptive method called BARS is shown in the overlay. It succeeds in capturing what we believe are the essential features of the firing rate curve. We next briefly outline what BARS is, and why we like it.



BARS uses cubic splines. A cubic spline is a piecewise cubic polynomial: several cubic pieces are joined at locations called "knots," shown at locations ζ_1, ζ_2 , and ζ_3 in the figure. The pieces are constrained so that the resulting curve is smooth (it is twice continuously differentiable). When we use a spline to fit a curve to some data we must begin by choosing how many knots to use and where these knots will go. This determines how smooth the curve will be in each interval of time. From a statistical point of view, choosing the knots is the hard part of the problem.

Bayesian Adaptive Regression Splines (BARS) DiMatteo, Genovese, Kass (*Biometrika*, 2001)

- Monte Carlo simulation method
- searches for number of knots and locations of knots
- produces optimally fitted curve and assessments of uncertainty

BARS, which stands for Bayesian Adaptive Regression Splines, was defined and studied in a recent paper by DiMatteo, Genovese, and Kass (*Biometrika*, 2001). In the statistical literature, this kind of application of splines is called "regression splines" because the fitting procedure uses linear regression and its generalizations. BARS uses a Bayesian Monte Carlo method to search through the space of possible numbers of knots and their locations, and to provide an optimally fitted curve based on this search. It also provides assessments of statistical uncertainty, which is crucial for the sorts of applications we outlined above.

For IT neuron: peak minus trough difference $50(\pm 20.8)$ spikes/sec greater for Condition 2 than for Condition 1

For example, in the comparison of distractor and non-distractor conditions in the IT neuron shown earlier DiMatteo *et al.* computed both an estimated difference in peak minus trough heights of 50 and also a standard error of 20.8 spikes per second to indicate uncertainty in the estimate.

	SARS	DMS	BARS
Example 1	0.144 (0.030)	0.206 (0.029)	0.061 (0.007)
Example 2	0.015 (0.001)	0.025 (0.002)	0.008 (0.001)
Example 3	0.044 (0.006)	0.106 (0.007)	0.018 (0.003)

DiMatteo *et al.* showed in a simulation study that BARS reduces the mean integrated squared error (MISE) of the fitted curve substantially compared to other methods in the literature. The table shows MISE (with standard error) for BARS along with two recently-published methods, which themselves were shown to work well compared to other available methods. BARS appears to be the best available method for adaptive fitting of smooth firing rates.

Poisson prob. spike at time t: $\lambda(t)dt$ **General** prob. spike at time t: $\lambda(t \mid s_1, \dots, s_k)dt$

 s_1,\ldots,s_k are spike times prior to t

We would like to mention, briefly, two other applications of spline-based smoothing. The first applies to situations where the non-Poisson nature of neural spiking activity may play an important role in data analysis. According to the Poisson process model the probability of a spike at time t is $\lambda(t)dt$. Please note that some people when they refer to a Poisson model mean a *constant, time-invariant* firing rate, but here we use the term in its more general *time-varying* or *inhomogeneous* form, which specifically allows for a time-varying firing rate (as we observe in experimental data). The beauty of the Poisson model is that the firing rate depends *only* on time. To generalize, we must allow the firing rate to depend also on past spikes, as shown in the second line here. This general form, however, is too general to be useful: we now have a λ function that depends not only on time t but also on the values of all the spike times prior to t, and to fit this firing rate empirically from the data is an impossibly difficult task.

```
Poisson prob. spike at time t: \lambda(t)dt
General prob. spike at time t: \lambda(t \mid s_1, \dots, s_k)dt
```

 s_1,\ldots,s_k are spike times prior to t

IMI prob. spike at time t: $\lambda(t, t - s_*(t))dt$ $s_*(t)$ is time of previous spike IMI stands for Inhomogeneous Markov Interval process

To simplify things Kass and Ventura (2001, *Neural Computation*) introduced Inhomogeneous Markov Interval (or IMI) processes. The simplification here is that the firing rate now depends on only two variables: time t, and the time since the last spike prior to t.

```
Poisson prob. spike at time t: \lambda(t)dt
General prob. spike at time t: \lambda(t \mid s_1, \dots, s_k)dt
```

 s_1, \ldots, s_k are spike times prior to t

IMI prob. spike at time t: $\lambda(t, t - s_*(t))dt$ $s_*(t)$ is time of previous spike IMI stands for Inhomogeneous Markov Interval process splines may be used to fit $\lambda(t, t - s_*(t))$

This allows the same kind of empirical fitting as we showed previously, using spline methods, now complicated in a manageable way by involving a function of two variables rather than one. Kass and Ventura noted that standard software could be used to fit this non-Poisson probability model.



The fit of various probability models may be displayed using a graphical method known as a *Q-Q plot*, as discussed by Brown, Barbieri, Ventura, Kass, and Frank (2001, *Neural Computation*). In the q-q plot, a statistically good fit of a model is indicated by a nearly diagonal curve that falls within a set of 95% bands. In this figure, taken from the Brown *et al.* paper, goodness-of-fit for the PSTH, the inhomogeneous Poisson, and the IMI model are all displayed. It may be seen that the Poisson model improves substantially on the PSTH, but the IMI model improves further on the Poisson and may be judged to fit the data well.

prob. of spike at time t for neuron 1: $p^1(t)$ prob. of spike at time t for neuron 2: $p^2(t)$ prob. of simultaneous spike at time t: $p^{12}(t)$

independence model: $p^{12}(t) = p^1(t)p^2(t)$

Our final application of spline-based smoothing is to the study of correlated firing in a pair of neurons. Here we examine the simultaneous firing of two neurons, with simultaneity defined as coincident spikes occurring within a specified window of time such as 1 millisecond. The statistical issues are, first, whether there is any evidence that the coincident firing is due to sources other than chance, i.e., whether the independence model, shown here, is inadequate and, second, if so, what the departures from independence may be.

prob. of spike at time t for neuron 1: $p^1(t)$ prob. of spike at time t for neuron 2: $p^2(t)$ prob. of simultaneous spike at time t: $p^{12}(t)$ independence model: $p^{12}(t) = p^1(t)p^2(t)$ replace independence model with

$$p^{12}(t)=p^1(t)p^2(t)\zeta(t)$$

We consider two possibilities. First we introduce a function $\zeta(t)$, which models the time-dependent change in the *simultaneous* firing rate *beyond* what would be predicted by the independence model.

prob. of spike at time t for neuron 1: $p^1(t)$ prob. of spike at time t for neuron 2: $p^2(t)$ prob. of simultaneous spike at time t: $p^{12}(t)$ independence model: $p^{12}(t) = p^1(t)p^2(t)$ replace independence with

$$p_i^{12}(t) = \alpha_i^1 \alpha_i^2 p^1(t) p^2(t) \zeta(t)$$

where i signifies i-th trial

In addition, we introduce constants representing global increments in each neuron's firing rate on particular trials. Thus, α_i^1 is a multiplicative increment (or decrement) of the firing rate for neuron 1 on the *i*th trial. The reason we add these coefficients is that we are thereby able to resolve two distinct features of coincident firing that are otherwise confounded in the joint PSTH and cross-correlogram: we can separate the effects that act to produce an excess (or deficit) in coincident spiking globally across time (the α_i 's) from those that produce an excess (or deficit) in a time-dependent fashion (the $\zeta(t)$ function). Having formalized the problem this way we may now apply spline-based methods to estimate $\zeta(t)$ from data.



We will illustrate with data simulated from a model that includes both a time-dependent $\zeta(t)$ function and coefficients α_i representing trial-to-trial variability. In this figure we show the firing rate functions for the two neurons, together with the $\zeta(t)$ function representing excess coincident spiking that we used in our simulation study. If $\zeta(t)$ were equal to the constant 1, the independence model would hold. The α_i values (not shown) were assumed equal for neuron 1 and neuron 2 and were generated from a Gamma distribution having mean 1.



We simulated 100 trials of data. In this figure we show the spline-fitted $\zeta(t)$ function from one of the data sets. It does a reasonably good job of uncovering the true time-varying excess in coincident spiking. Here the thick line is the true curve $\zeta(t)$ and the thin line is the fitted curve.



It is important that we included estimates of the α_i coefficients in our fitting method (the details of which we are omitting here). If we fail to do so, we get the dashed curve shown here, which is too high. We are trying to make two points here. First, a simple probability model can separate the effects of trial-related increase in neuronal activity from time-dependent coincident spiking. Second, spline-based fitting may be used to estimate these effects from data. Using our spline-based fits we can obtain confidence intervals or evaluate whether the effects we see are likely to occur due to chance alone.



In this last slide we show 95% simulation bands for $\zeta(t)$ under the null hypothesis that $\zeta(t)$ is equal to 1, i.e., that there is no time-dependent increase in coincident firing above what is implied by the independence model together with any global (time-invariant) increase or decrease in firing rates on particular trials. The fitted curve clearly



falls outside these bands for some values of t, which indicates that there is an increase in coincident firing over an interval of time. These bands were obtained by bootstrap methods. In addition, found that this degree of variation above the simulation bounds would occur by chance with a probability of p < .0005. Although the data were simulated, our intention here is to illustrate the sort of results that may be obtained by spline-based fitting combined with modern statistical inference methods such as the bootstrap.

- We estimate instantaneous firing rate because
 - * questions of interest often require instantaneous firing rate;
 - * probability models involving instantaneous firing rate make efficient use of the data.

- We estimate instantaneous firing rate because
 - * questions of interest often require instantaneous firing rate;
 - * probability models involving instantaneous firing rate make efficient use of the data.
- As an estimate of firing rate the PSTH can be greatly improved by smoothing.

- We estimate instantaneous firing rate because
 - * questions of interest often require instantaneous firing rate;
 - * probability models involving instantaneous firing rate make efficient use of the data.
- As an estimate of firing rate the PSTH can be greatly improved by smoothing.
- Splines are easy to use, but the trick is to pick the number and locations of the knots; BARS is currently the best available method

- We estimate instantaneous firing rate because
 - * questions of interest often require instantaneous firing rate;
 - * probability models involving instantaneous firing rate make efficient use of the data.
- As an estimate of firing rate the PSTH can be greatly improved by smoothing.
- Splines are easy to use, but the trick is to pick the number and locations of the knots; BARS is currently the best available method
- In addition to smoothing the PSTH, spline-based methods are useful for modeling
 - * non-Poisson spike trains;
 - * correlated firing among pairs of neurons.

- We estimate instantaneous firing rate because
 - * questions of interest often require instantaneous firing rate;
 - * probability models involving instantaneous firing rate make efficient use of the data.
- As an estimate of firing rate the PSTH can be greatly improved by smoothing.
- Splines are easy to use, but the trick is to pick the number and locations of the knots;
 BARS is currently the best available method
- In addition to smoothing the PSTH, spline-based methods are useful for modeling
 - * non-Poisson spike trains;
 - * correlated firing among pairs of neurons.
- BARS has also been applied to fMRI and EEG analysis.

- We estimate instantaneous firing rate because
 - * questions of interest often require instantaneous firing rate;
 - * probability models involving instantaneous firing rate make efficient use of the data.
- As an estimate of firing rate the PSTH can be greatly improved by smoothing.
- Splines are easy to use, but the trick is to pick the number and locations of the knots;
 BARS is currently the best available method
- In addition to smoothing the PSTH, spline-based methods are useful for modeling
 - * non-Poisson spike trains;
 - * correlated firing among pairs of neurons.
- BARS has also been applied to fMRI and EEG analysis.
- BARS is computationally-intensive. We are working on faster methods that retain desirable features.

REFERENCES

Baker, C.I., Behrmann, M., and Olson, C.R. (2001) Effect of learning representation of multipart objects in monkey inferotemporal cortex. *Soc. Neurosci. Abstr.*

Brown, E.N., Barbieri, R., Ventura, V., Kass, R.E., and Frank, L.M. (2001) The time-rescaling theorem and its application to neuronal spike train data analysis, *Neural Comput.*, 13, to appear.

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

Kass, R.E., and Ventura, V. (2001) A spike-train probability model. *Neural Comput.*, 13: 1713-1720.

Olson, C.R., Gettner, S.N., Ventura, V., Carta, R. and Kass, R.E. (2000) Neuronal activity in macaque supplementary eye field during planning of saccades in response to pattern and spatial cues. *J. Neurophys.*, 84: 1369-1384.

Olson, C.R. and Rollenhagen, J.E. (1999) Low-frequency oscillations in Macaque IT cortex during competitive interactions between stimuli *Soc. Neurosci. Abstr.*, 25: 916.

Ventura, V., Carta, R., Kass, R.E., Gettner, S.N., and Olson, C.R. (2001) Statistical analysis of temporal evolution in single-neuron firing rates. *Biostatistics*, to appear.