Estimating Neural Firing Rates from Single Trial Spike Trains via Machine Learning Optimized Kernel Smoothing

Taylor C. Hansen and W. Mitchel Thomas

Abstract— **Many neural decoding systems rely on real-time calculations of the neural firing rate to provide control for neural prostheses. Because the firing rate is variable with time and across trials, it is desirable to rapidly estimate the underlying firing rate function from a single trial. In this work, we implement a kernel convolution technique to estimate the underlying firing rate function of a neuron from a single spike train. We also explore an optimization technique using a simple perceptron to derive the best kernel parameters for firing rate estimates of various cell types. We find that these optimal parameters are dependent upon the underlying rate function and are similar to those derived in the literature from *ad hoc* methods.**

I. Introduction

A prevailing view in computational neuroscience is that neural spike trains are generated by an underlying firing rate which, in turn, is critical for understanding the encoding of neural information. Because spike trains from a given neuron can vary considerably between trials, knowledge about individual spike times has not proven as useful as this underlying firing rate. Unfortunately, the firing rate cannot be recorded directly and must instead be estimated by calculations on the resulting spike trains. Historically, peri-stimulus time histograms have been used as a way of obtaining an average firing rate, calculated over multiple, single-neuron recordings [1]. However, for real-time systems, repeated-trial designs are not always feasible nor desirable. As an alternative, various methods for inferring the underlying firing rate function from spiking activity during a single trial have been developed. One such method is kernel smoothing, in which the spike train is convolved with a kernel to generate a smooth estimate of the firing rate function [2]. This approach is amenable to online applications, provided that the kernel width and shape have been properly selected. These attributes are often selected in an *ad hoc* way [3]. To further improve the kernel smoothing firing rate estimate, we propose that optimization of kernel width and shape can be accomplished using a simple perceptron. We further postulate that these optimal hyperparameters will vary between cell types. We find that the simple perceptron is able to rapidly determine both the optimal kernel width and kernel shape for use with a single-trial spike train [4]. With this improved method of firing rate estimation, we show the feasibility of one potential application in the realm of neural prostheses.

II. Methods

A. Rate Functions and Spike Train Simulations

In order to create spike trains for training and testing of machine learning algorithms, we utilized a set of equations for the generation of rate functions and spike trains [2].

$$\beta(t) := \begin{cases} \frac{1}{\tau_2-\tau_1}\left(e^{-\frac{t}{\tau_1}} - e^{-\frac{t}{\tau_2}}\right) & for\ t > 0 \\ 0 & for\ t < 0 \end{cases} \quad (1)$$

where $\tau_2 > \tau_1 > 0$ are falling and rising time constants. This function is normalized to obtain a unit area. The response width or length of a given train was given by equation (2).

$$w = \sqrt{\tau_1^2 + \tau_2^2} \quad (2)$$

The final rate functions are given by the equation

$$\rho(t) := b + \mathrm{A}\beta(\mathrm{t} - \mathrm{t}_0) \quad (3)$$

where b is the background rate, A is the response strength, and t_0 corresponds to the first observed non-stochastic spike.

Spike trains were simulated by creating a Poisson point process. This Poisson process was made by pulling from one of three rate functions and an exponential distribution over a set time interval. The rate functions were created using 20, 30, and 40 Hz background rates with response widths 80, 110, and 140 ms to mimic types of neurons with different firing rates.

B. Kernel generation

To estimate the underlying neural firing rate function, the resulting variable spike trains generated from the Poisson point process were convolved with a fixed kernel according to the methodology described in detail by Nawrot [2]. The estimate of the firing rate function is given by equation (4):

$$\lambda(t) = \sum_{t=1}^{n} K(t - t_i) \quad (4)$$

where t_i is the vector of spike occurrence times and $K(t - t_i)$ is the zero-mean kernel function centered at t_i. Two kernel characteristics were varied before convolution: shape and width. Four kernel shapes were tested: boxcar, triangle, Epanechnikov, and Gaussian, as described and bounded by Nawrot [2]. A suite of kernel widths (σ) was tested, with values determined by the simple perceptron described below. Each

T. C. Hansen is with the Biomedical Engineering Department, University of Utah, Salt Lake City, UT 84132 USA (e-mail: taylor.c.hansen@utah.edu).

W. M. Thomas is with the Biomedical Engineering Department, University of Utah, Salt Lake City, UT 84132 USA (e-mail: wm.thomas@utah.edu).

kernel was limited to non-zero values and was normalized to give each kernel unit area.

C. Simple perceptron

A perceptron was used to estimate the optimal kernel width for a set of spike trains generated from 20, 30, and 40 Hz ideal rate functions with corresponding widths 80, 110, and 140 ms. The perceptron consisted of two nodes representing bias and the tuning of a width scale (γ) hyperparameter. The width scale parameter was set to 2.5 for all trials. A single trial consisted of 30 iterations. For each iteration, the perceptron convolved a randomly generated spike train from a known ideal rate function for each kernel shape. The integrated square error (ISE) was used as the error metric, which calculates the difference between the estimated rate function and the ideal rate function:

$$ISE = 2\int_0^T \left(K\left(\sigma_{\theta_i}(\gamma), t\right) - \lambda(t)\right)^2 dt \quad (5)$$

where $K(\sigma_\theta(x_{ws}), t)$ is the kernelized estimate of the spike rate function, $\sigma_\theta(x_{ws})$ is the kernel width determined via linear regression of the perceptron output, γ is the width scaling hyperparameter and $\lambda(t)$ is the ideal rate function.

The ISE was used as the cost function, where the goal of the perceptron was to minimize the ISE with respect to the perceptron weights. To perform this, the gradient of the ISE was used for a perceptron update:

$$\theta_{1,i+1} = \theta_{1,i} - \eta\left(\frac{\partial ISE}{\partial \theta_i} - \frac{\partial ISE}{\partial \theta_{i-1}}\right) \quad (6)$$

$$\theta_{1,i+1} = \theta_{1,i} - \eta\left(\frac{\partial ISE}{\partial \theta_i} - \frac{\partial ISE}{\partial \theta_{i-1}}\right)\gamma$$

where the gradient is defined as:

$$\frac{\partial ISE}{\partial \theta_i} = \int_0^T K\left(\sigma_{\theta_i}(\gamma), t\right) - \lambda(t)\, dt \quad (7)$$

and $\frac{\partial ISE}{\partial \theta_{i-1}}$ is the gradient of the previous iteration.

At the end of each training, the width with the lowest ISE was selected as the optimal kernel width, as the minimization of our cost function did not consistently converge to a global minimum after 30 trials.

All coding was done in MATLAB.

III. Results

A. Varying underlying firing rate

The first set of simulations examined the estimated firing rate function after spike train convolution with triangular kernels (Fig. 1). Three underlying firing rate functions were simulated and estimated with this method, each corresponding to a column in (Fig. 1). Higher underlying firing rate functions (Fig. 1A) were associated with more accurate estimates with lower underlying rate functions being more poorly described by the kernel-convolved estimates.

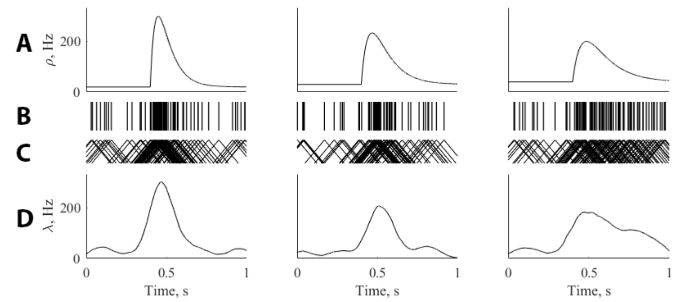

Figure 1 - Varying underlying firing rate for estimation by convolution with triangular kernels. (A) Three underlying firing rate functions were simulated. From left to right: b = 20 Hz, w = 80 ms; b = 30 Hz, w = 110 ms; b = 40 Hz, w = 140 ms. (B) An example spike train generated from each of the underlying rate functions. (C) The associated triangular kernels are shown below each spike train with σ = 50 ms. (D) Finally, the estimated firing rate functions, as a function of time, are shown.

B. Varying kernel width

Next, simulations were run to observe the influence of kernel width (σ) on the firing rate estimate (Fig. 2). All estimates were done using the same spike train (Fig. 2B) derived from an underlying firing rate function with b = 20 Hz and w = 100 ms (Fig. 2A). Smaller σ appeared associated with a more variable estimated rate, whereas increased σ smoothed the estimate considerably. These results highlight the need for optimizing this parameter given the variability in the estimated firing rate function from an identical spike train.

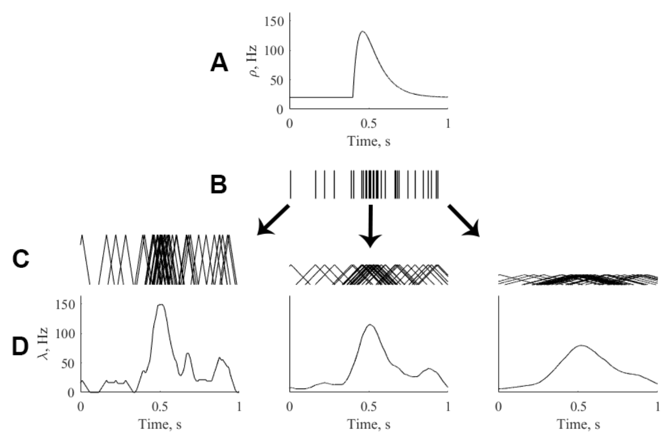

Figure 2 - Varying kernel width (σ) of triangular kernels. (A) The underlying firing rate function with b = 20 Hz and w = 100 ms. (B) The same spike train was used for all estimates. (C) Three kernel widths (σ) were used. From left to right: 20 ms, 50 ms, and 100 ms. Unit area under each kernel was maintained when varying σ. (D) Resulting estimates of the firing rate function. Increasing σ is associated with increased smoothening of the estimate.

C. Varying kernel shape

Four distinct kernel shapes were investigated to determine the impact on the firing rate function estimate (Fig. 3). All four kernel shapes were assessed on the same spike train (Fig. 3B). A σ of 50 ms was used for all four kernel shapes. The resulting estimates are qualitatively quite similar, with the boxcar generating the roughest estimate of the four (Fig. 3G). The other three kernel estimates are hardly distinguishable.

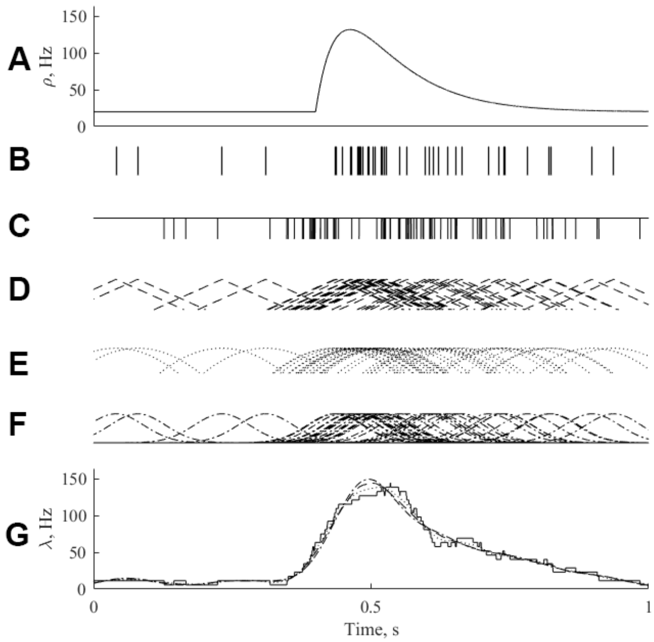

Figure 3 – The effects of varying kernel shape on the estimate of the firing rate function. (A) The underlying firing rate function with b = 20 Hz and w = 100 ms. (B) The same spike train was used for all estimates. (C) The boxcar kernels. (D) The triangular kernels. (E) The Epanechnikov kernels. (F) The Gaussian kernels. (G) The estimated firing rate functions for each kernel shape are overlaid on the same plot. Only the boxcar kernel appears to have a noticeably different estimate, although all four rates are qualitatively similar.

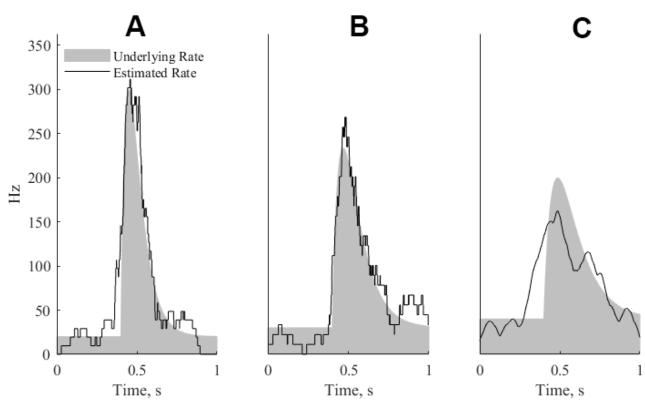

Figure 4 – Typical firing rate function estimates from perceptron-derived optimal kernel parameters. (A) An underlying firing rate function with b = 20 Hz and w = 80 ms (shaded gray). The optimal estimated rate function in this case is derived from a boxcar kernel with σ = 29.7 ms (solid overlay). (B) An underlying firing rate function with b = 30 Hz and w = 110 ms (shaded gray). The optimal estimated rate function in this case is derived from a boxcar kernel with σ = 25.8 ms (solid overlay). (C) An underlying firing rate function with b = 40 Hz and w = 140 ms (shaded gray). The optimal estimated firing rate function in this case is derived from an Epanechnikov kernel with σ = 33.3 ms (solid overlay).

D. *Optimizing Kernel Parameters with Simple Perceptron*

Following the exploratory simulations above, three simple perceptrons were generated to find the optimal kernel shape and width for estimating a given underlying firing rate function. Some typical estimations using the optimally derived parameters on a single spike train are shown in (Fig. 4). The perceptron frequently determined that the boxcar was the most accurate shape, although occasionally the Epanechnikov was optimal. In over 100 optimizations, we did not observe cases where triangular or Gaussian kernels were determined by the perceptron to be the optimal shape. The optimal σ varied depending on the optimal shape as well as the underlying rate function but was usually somewhere between 20 and 40 ms. As the underlying firing rate decreased, the estimates tended to not be as accurate (Fig. 4C).

IV. Discussion

A. *Variations of Kernel Parameters*

Our initial simulations served as an important milestone for understanding the influence and scope of the kernel parameters as well as the variability introduced by the underlying firing rate function. Varying this underlying rate showed that the kernel estimate performed well when the peak underlying firing rate was relatively high (e.g. above 200 Hz). It was also observed that the estimate performed more poorly as the underlying rate function was lower and wider. Varying the kernel width (σ) showed that this parameter's main effect was to smooth the estimate of the underlying rate function. This parameter was ultimately more influential than the kernel shape in determining how accurate the estimate would be. For this reason, σ was an enticing parameter over which to optimize with a machine learning approach.

B. *Perceptron Performance for Optimizing Kernel Width*

The final output of the perceptron found optimal σ's between 20 ms and 30 ms, which is a similar range of optimal σ's reported in previous studies [5]. While better kernel fits were found for smaller ideal firing frequencies, the perceptron yielded adequate fits for all frequencies tested. However, the mathematical means with which the perceptron obtained these optimal σ's was not well defined and did not keep with standard machine learning practices. A standard perceptron update uses only the gradient of the cost function during the current time step. Our perceptron utilized the difference between the gradient of the previous and current spike train tested. If only the current time step was used, the ISE cost function did approach a minimum value, but the optimal σ's were found to be over 100 ms. This created a flat kernel fit and did not match results found in previous studies. The difference in gradient update rule did not minimize the ISE cost as well as the sole gradient update rule, but the minimum value of the ISE during the training period always yielded an optimal σ within the range seen in prior studies, and a subsequent good kernel fit. Due to time constraints, we ran with this methodology as it continued to supply consistent, good results. More work needs to be done to tease out why the sole gradient update rule failed to accurately find the best σ for the convolution and why large kernel widths minimized the ISE.

REFERENCES

[1] G. L. Gerstein and N. Y.-S. Kang, "An approach to the quantitative analysis of electrophysiological data from single neurons," *Biophys. J.*, vol. 1, pp. 15-28, 1960.

[2] M. Nawrot, A. Aertsen, and S. Rotter, "Single-trial estimation of neuronal firing rates: from single-neuron spikes to population activity," *J. Neurosci. Methods*, vol. 94, pp. 81-92, 1999.

[3] J. P. Cunningham, V. Gilja, S. I. Ryu, and K. V. Shenoy, "Methods for estimating neural firing rates, and their application to brain-machine interfaces," *Neural Networks*, vol. 22, pp. 1235-1246, 2009.

[4] Rosenblatt, Frank. The perceptron: A probabilistic model for information storage and organization in the brain [J]. Psychol. Review. 1958;65:386 - 408.

[5] Shimazaki H, Shinomoto S. Kernel bandwidth optimization in spike rate estimation. J Comput Neurosci. 2009;29(1-2):171–182.