Bayesian decoding using unsorted spikes in the rat hippocampus

Fabian Kloosterman,1,2,3,4,5 Stuart P. Layton,1,2 Zhe Chen,1,2,6 and Matthew A. Wilson1,2

1Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, Massachusetts; 2Picower Institute for Learning and Memory, Massachusetts Institute of Technology, Cambridge, Massachusetts; 3NERF, Leuven, Belgium; 4imec, Leuven, Belgium; 5Laboratory of Biological Psychology, Department of Psychology, KU Leuven, Leuven, Belgium; and 6Neuroscience Statistics Research Lab, Massachusetts General Hospital, Harvard Medical School, Boston, Massachusetts

Submitted 3 December 2012; accepted in final form 30 September 2013

Kloosterman F, Layton SP, Chen Z, Wilson MA. Bayesian decoding using unsorted spikes in the rat hippocampus. J Neurophysiol 111: 217–227, 2014. First published October 2, 2013; doi:10.1152/jn.01046.2012.—A fundamental task in neuroscience is to understand how neural ensembles represent information. Population decoding is a useful tool to extract information from neuronal populations based on the ensemble spiking activity. We propose a novel Bayesian decoding paradigm to decode unsorted spikes in the rat hippocampus. Our approach uses a direct mapping between spike waveform features and covariates of interest and avoids accumulation of spike sorting errors. Our decoding paradigm is nonparametric, encoding model-free for representing stimuli, and extracts information from all available spikes and their waveform features. We apply the proposed Bayesian decoding algorithm to a position reconstruction task for freely behaving rats based on tetrode recordings of rat hippocampal neuronal activity. Our detailed decoding analyses demonstrate that our approach is efficient and better utilizes the available information in the nonsortable hash than the standard sorting-based decoding algorithm. Our approach can be adapted to an online encoding/decoding framework for applications that require real-time decoding, such as brain-machine interfaces.

neural decoding; spatial-temporal Poisson process; population codes; kernel density estimation; spike sorting

FEATURES OF SENSORY STIMULI and intended motor actions are reflected in the activity of neuronal ensembles that are distributed across the brain (Boloori et al. 2010; Huys et al. 2007; Sanger 2003). A fundamental goal in neuroscience is to understand how the information about external stimuli is transformed into neural activity patterns and how this information is represented in the brain. The relationship between stimuli and neural activity can be described by statistical encoding models (Brown et al. 1998; Paninski et al. 2007; Sanger 2003; Truccolo et al. 2005). Inversion of these encoding models, i.e., extraction of information about the stimulus from observed neural activity (“neural decoding”), aids in revealing the principles of the encoding process (Quiña Quiroga and Panzeri 2009). Neural decoding can also be applied to uncover internal neuronal representations in the absence of an overt stimulus, for example, the reexpression of spatial sequences in the hippocampus (Davidson et al. 2009; Gupta et al. 2010) or movement intentions in motor cortices (Georgopoulos et al. 1986; Zhuang et al. 2010). Decoding motor plans in particular is an important part of the development of neural prosthetics and brain-machine interfaces that may restore motor function in patients with neurological damage (Chapin 2004; Hochberg et al. 2012; Schwartz et al. 2006).

The principal goal of neural decoding is to extract as much information about a stimulus as possible from a neural signal. As with all signal processing, any additional operation on the raw signal in neural decoding adds complexity and leads to possible loss of information. Most approaches for decoding neural spiking activity rely on the intermediate step of sorting spike waveforms into groups of single units. The spike sorting process is subject to at least two problems that could affect decoding performance. The first issue is that the goal of neural decoding to minimize decoding error is substantially different from the goal to label each spike uniquely and confidently with the identity of the cell that emitted it. In particular, spike sorting is generally conservative, and many spikes are left unclassified in an attempt to minimize classification errors. However, spikes thrown out during the sorting process could still convey information about the stimulus and hence contribute to decoding performance. Second, inherent to spike sorting are misclassification errors, that is, incorrect assignment of spikes to a unit. Theoretical analysis has shown that different spike sorting errors have various impacts on information capacity, with false positive errors having the most serious effect (Goodman and Johnson 2008). Another potential source of misclassification is the use of hard decision boundaries—it has been suggested that a soft decision boundary is more appropriate for evaluating neural ensemble codes (Wood and Black 2008).

To maximize the stimulus information extracted from neural spiking activity, we propose a novel Bayesian decoding paradigm that does not require the intermediate step of spike sorting. Key in our approach is a direct mapping between the raw data (i.e., spike waveform features) and the stimulus in a joint probability distribution. In contrast to previous work, our approach does not assume a parametric or biophysical model to describe the relation between stimulus and neural activity.

The performance of the new decoding approach is analyzed by applying it to hippocampal population recordings in order to estimate the location of a rat on a track.

METHODS

Spike Feature Decoding Framework

Encoding model. The ultimate goal of our method is to reconstruct a sensory stimulus, motor action, or other covariate (from here on referred to as “stimulus”) from neuronal spiking activity recorded from an array of sensors (e.g., single-wire electrodes, stereotrodes, or...
tetrodes). First, we build an encoding model that relates the neural activity on a single sensor to the stimulus of interest. Let us assume that in the time period \((0, T]\) we recorded the time-varying stimulus vector \(x(t)\) as well as \(N\) discrete spike events and their waveforms at times \(t_n\), with \(0 < t_1 < t_2 < \ldots < t_N \leq T\). For simplicity we assume that all spikes occur conditionally independent of previous spikes and treat the detected spikes as a spatio-temporal Poisson process, or equivalently as a marked temporal Poisson process, in which the spatial component (the “mark”) is a vector space \(S\) of spike waveform features \(a \in S\). Examples of typical waveform features are peak amplitude, spike width, extracted principal components (PCs), or other derived features. The same waveform features are generally used by spike sorting processes to extract single units from multiunit activity (MUA). In our approach the spike sorting step is bypassed by creating a direct mapping between spike waveform features and stimulus (Fig. 1). The spatio-temporal Poisson process is fully characterized by its generalized rate function \(\lambda(a, t)\). In cases in which the rate is determined by the stimulus of interest, the rate function can be reexpressed as \(\lambda(a, t) = \lambda(a, x(t))\). Here, \(\lambda(a, x)\) can be viewed as a tuning curve that relates the average rate of spike events with waveform features \(a\) to the stimulus \(x\).

To compute the probability that we observe \(n\) spikes with associated features \(a_1: n\) in a small time window \([t, t + \Delta t]\) in the presence of a known stimulus, the waveform feature space \(S\) is first divided into ...

Fig. 1. A: diagram of spike waveform feature decoding approach. In the encoding stage (top), a model describing the relation between spike waveform features and the stimulus of interest (for example, an animal’s position) is constructed. Note that no prior sorting of the recorded spikes is required. In the decoding stage (bottom), the encoding model is used to obtain an estimate of the stimulus from the measured spike waveform features. B: pictographic overview of major steps in the algorithm of the spike feature decoding approach. From spike waveform features computed for the recorded raw spike waveforms and the known stimulus, the probability distributions \(p(a, x), p(x),\) and \(\pi(x)\) are derived. From these distributions, the joined rate function \(\Lambda(a, x)\) and marginal rate function \(\lambda(x)\) can be derived. During the decoding process, observed spike waveform features in single time bins are used to evaluate the rate functions [indirectly through evaluation of the probability distributions \(p(a, x), p(x),\) and \(\pi(x)\)] at a user-defined stimulus grid. The evaluated rates are used to compute the posterior distribution.
J nonoverlapping regions: \( S = \{S_1 \cup S_2 \cup \ldots \cup S_J\} \). Each region \( S_j \) contains \( n_j \) spikes, which is a subset of the observed \( n \) spikes (i.e., \( \sum_{j=1}^{J} n_j = n \)). The expected number of spikes in each region \( S_j \) follows a Poisson distribution with rate function \( \lambda_j(x) = \int x \lambda(a,x) \, da \).

The likelihood of finding exactly \( n_{1,j} \) spikes in regions \( S_{1,j} \) can be computed from the product of Poisson likelihoods of all regions:

\[
P(a_{1,j} | x) = P(n_{1,j} | x) = \prod_{j=1}^{J} P(n_{1,j} | x) = \prod_{j=1}^{J} \text{Poisson}(n_{1,j}; \lambda_j(x))
\]

\[
= \left[ \prod_{j=1}^{J} (\frac{\lambda_j(x)^{n_{1,j}}}{n_{1,j}!}) e^{-\lambda_j(x)} \right]^{n_{1,j}}
\]

In the limiting case when regions \( S_{1,j} \) become sufficiently small such that \( n_{1,j} \) are equal to 0 or 1 within the time interval \( \Delta t \), the likelihood can be rewritten as

\[
P(a_{1,1} | x) = \left( \frac{\Delta t}{\Delta x} \right)^{n_{1,1}} \left[ \lambda(a, x) \right]^{n_{1,1}} e^{-\lambda(a, x)}
\]

This likelihood function, when calculated for all possible waveform feature vectors \( a \), completely characterizes the encoding process for a single sensor. For multiple sensors, assuming conditional spiking independence between sensors (i.e., each sensor records from an independent population of neurons), the joint data likelihood can be computed as a product of individual likelihoods contributed by each sensor. For \( K \) sensors and \( n_j \) spike events on the \( k \)th electrode, the joint likelihood is given by

\[
P(a^{1:K} | x) = \prod_{k=1}^{K} P(a_{1,1} | x)
\]

### Relation to encoding with spike-sorted units or multunit activity

It is possible to choose the spike feature such that it is a discrete scalar variable that represents cell identity (for example, obtained through a spike sorting procedure). In that case, each region \( S_j \) can be constructed such that it corresponds to a single cell \( c \). This means that \( \lambda_j(x) = \lambda_j(x) \) is the tuning curve of cell \( c \) and \( n_j = n_j \) is the number of spikes emitted by cell \( c \). By rewriting Eq. 1 we recover the likelihood for a population of spike-sorted cells as a special case (Zhang et al. 1998):

\[
P(n_{1,c} | x) = \left( \frac{\Delta t}{\Delta x} \right)^{n_{1,c}} \left[ \lambda_j(x) \right]^{n_{1,c}} e^{-\lambda_j(x)}
\]

where \( C \) is the total number of cells and \( N = \sum_{c=1}^{C} n_c \) is the total number of spike events.

When all spikes are considered part of a single multunit cluster and spike features \( a \) are ignored, then the likelihood can be further simplified to

\[
P(N | x) = \left( \frac{\Delta t}{\Delta x} \right)^N \left[ \lambda(x) \right]^N e^{-\lambda(x)}
\]

### Evaluation of the likelihood

To compute the likelihood in Eq. 2, representations of the generalized rate function \( \lambda(a, x) \) and its marginal rate function \( \lambda(x) \) need to be constructed. A parametric model of the rate functions would allow for straightforward evaluation during decoding; however, it is not clear what model, if any, would be appropriate. In contrast, nonparametric models provide a more flexible estimate of the rate functions. Here we used kernel density-based estimators of the rate functions.

To construct the kernel density estimators, the generalized rate function \( \lambda(a, x) \) and the marginal rate function \( \lambda(x) \) are decomposed into spike event probability distributions \( p(a, x) \) and \( p(x) \) and a stimulus probability distribution \( p(x) \):

\[
\lambda(a, x) = \frac{\text{spikecount}(a, x)}{\text{occupancy}(x)} = \frac{N p(a, x)}{T \pi(x)} = \mu p(a, x) \pi(x)
\]

Bayes’ rule provides a way to combine prior information about the stimuli \( \lambda(x) \) with information obtained from observations through the likelihood function \( P(a_{1,m} | x) \) (taken from Eqs. 2 and 3). The denominator \( P(a_{1,m}) \) is a normalizing constant such that the posterior \( P(x | a_{1,m}) \) is a proper probability distribution. The posterior distribution contains all information about the stimulus at time \( t \) that can be extracted from the observed spike events. If a noninformative temporal prior is assumed, then the aim of Bayesian decoding is to maximize the product of the likelihood and time-independent prior: \( P(x | a_{1,m}) \propto P(a_{1,m}) P(x) \). If a completely noninformative prior is assumed, then Bayesian decoding is analogous to maximum likelihood estimation: \( P(x | a_{1,m}) \propto P(a_{1,m}|x) \).

### Implementation

The spike feature-based decoder was implemented in MATLAB (The MathWorks, Natick, MA) with custom C exten-
Innovative Methodology

BAYESIAN DECODING USING UNSORTED SPIKES

sions. In Eqs. 6–10 we developed a representation for the rate functions \( \Lambda (a(x)) \) and \( \Lambda (x) \), which are needed to compute the likelihood in Eq. 2. The rate functions either can be precomputed at a user-defined grid to construct look-up tables or can be evaluated online during decoding. Precomputing the rate functions has the advantage that the computational load for decoding is low, making it suitable for real-time applications. When densities need to be evaluated at a fine grid for high-dimensional feature vectors and/or stimulus vectors, then the precomputing approach becomes impractical. In that case, the rate functions may be evaluated at a coarser grid, at the expense of decreasing accuracy. Instead, here we chose to evaluate the rate function \( \Lambda (a(x)) \) on the fly during decoding at a user-defined grid in stimulus space. \( \Lambda (x) \) was precomputed at the same stimulus grid. A small offset (0.1 Hz) was applied to both rate functions to avoid cases in which the rate is zero at all nodes in the stimulus grid (Davidson et al. 2009; Zhang et al. 1998). Decoding is performed separately for predefined time bins. For each detected spike and its associated features within a time bin, the rate function \( \Lambda (a(x)) \) (Eq. 6) is evaluated, after which the likelihood (Eqs. 2 and 3) and the posterior distributions (Eq. 11) are computed. Computations are performed in log-space, such that products in Eq. 2 are replaced by summations.

Application to Hippocampal Recordings

Electrophysiology and behavior. Eight male Long-Evans rats, weighing between 400 and 600 g, were implanted with custom-made micro-drive arrays (Kloosterman et al. 2009; Nguyen et al. 2009). Individual arrays carried between 9 and 24 independently movable tetrodes targeted either to the right dorsal hippocampus or bilaterally to both hippocampi (coordinates: 2.5 mm lateral and 4.0 mm posterior to bregma). The tetrodes were slowly lowered into the brain until they reached the cell layer of CA1 2–4 wk after array implantation. A reference electrode was positioned in the white matter overlying the dorsal hippocampus. In the case of bilateral recordings, separate ipsilateral references were used for each hippocampus. Signals were filtered between 300 Hz and 6 kHz, and all extracellular spike waveforms that crossed a preset amplitude threshold (75 \( \mu \text{V} \)) on any of the four tetrode channels were sampled at 31,250 Hz (32 samples per spike waveform) and saved to disk.

During 30-40-min-long recording sessions rats were allowed to freely explore an ~3-m-long linear or circular track. While no behavioral restrictions were placed on the rats, they did receive a small food reward (chocolate sprinkle, fruit loop, etc.) to encourage exploration. The position of the rats was tracked at 30 Hz with an overhead video camera and infrared light-emitting diodes mounted on the implanted micro-drive array. All experiments were conducted under protocols reviewed and approved by the Massachusetts Institute of Technology Committee on Animal Care and followed the guidelines of the National Institutes of Health.

Decoding of animal’s position. Data collected in a recording session were divided into a training set for construction of the encoding model and a testing set for evaluation of the decoding. In both data sets only RUN epochs, when the animals were actively moving along the track at a speed higher than 10 cm/s, were selected for further analyses. The stimulus of interest \( x \) corresponds to the one-dimensional position of the animals along the track. Only spikes of putative pyramidal cells (spike peak–trough latency > 0.35 ms) with a minimum peak amplitude of 75 \( \mu \text{V} \) were selected for decoding analysis. A four-dimensional spike feature vector \( a \) was constructed from the spike waveform peak amplitudes on each tetrode. To establish a kernel density-based estimate of the rate functions we used a truncated Gaussian kernel (cutoff at 2 standard deviations) for both the spike amplitude dimensions and the position dimension. For spike amplitude an isotropic kernel was used with the same kernel bandwidth in all four dimensions. The rate functions were evaluated at a regular grid with 2-cm intervals that spanned the whole track. For decoding, the testing data set was divided into nonoverlapping 250-ms-long time bins and spikes within each bin were used to compute the posterior distribution of position according to Eq. 11.

In the Bayesian decoding framework, the decoding performance is determined by both the choice of encoding model and prior knowledge of the stimulus. Here we focus on the contribution of the novel formulation of the encoding model that includes spike waveform features, and hence we chose to use a noninformative prior, making the decoding similar to maximum likelihood estimation.

The decoding error in each time bin was computed as the shortest distance along the track between the true (observed) position at the center of the time bin and the maximum a posteriori (MAP) estimate of position. To assess decoding performance, we analyzed the cumulative distribution of decoding error, the median error, and the confusion matrix.

Spike sorting-based decoding. Spikes of putative pyramidal neurons (spike peak–trough latency > 0.35 ms, minimum peak amplitude > 75 \( \mu \text{V} \)) in RUN epochs were selected prior to sorting. Spikes were automatically sorted with KlustaKwik (version 2.0.1, http://klustakwik.sourceforge.net/; see Harris et al. 2000) using three features per electrode: spike waveform energy and the first two PCs of the energy-normalized spike waveform. For each cluster a set of four quality metrics was computed: Lratio, and isolation distance (IsoD; using waveform energy and 1st PC for each channel; see Schnitzer-Torbert et al. 2005), fraction of interspike intervals shorter than 2 ms (ISI), and the Mahalanobis distance of the cluster to the spike amplitude threshold (Dthreshold; using spike amplitudes on the electrode with largest mean spike amplitude). Only clusters that met the following criteria were retained: Lratio < 0.1, ISO > 20, ISI < 2%, and Dthreshold > 1. The remaining clusters were collapsed into a single noise or “hash” cluster. The spatial tuning curve of each cluster was constructed by kernel-density estimation with a truncated Gaussian kernel with an optimized spatial bandwidth, and the likelihood was computed according to Eq. 4.

Bandwidth selection. We used a diagonal bandwidth matrix to compute the kernel density estimates in Eqs. 8–10. Since the spike features that were used for decoding all represent the same physical quantity (i.e., spike peak amplitude) with a similar distribution, the same scalar bandwidth (\( h_s \)) was selected for all the feature dimensions and for all tetrodes. A separate bandwidth (\( h_n \)) was chosen for the stimulus dimensions (i.e., position on track). For each separate data set, optimal values for \( h_s \) and \( h_n \) were determined by a twofold cross-validation procedure on the training set.

RESULTS

A total of 12 data sets from 8 rats were used to test the spike feature-based decoding approach. Table 1 summarizes the

<table>
<thead>
<tr>
<th>Data Set</th>
<th>No. of Tetrodes</th>
<th>No. of Total Spikes</th>
<th>No. of Sorted Spikes</th>
<th>% Sorted Spikes</th>
<th>No. of Clusters</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1D2</td>
<td>14</td>
<td>81 K</td>
<td>5 K</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>R2D2</td>
<td>18</td>
<td>106 K</td>
<td>7 K</td>
<td>6</td>
<td>12</td>
</tr>
<tr>
<td>FK11</td>
<td>9</td>
<td>195 K</td>
<td>15 K</td>
<td>8</td>
<td>18</td>
</tr>
<tr>
<td>R1D1</td>
<td>14</td>
<td>170 K</td>
<td>36 K</td>
<td>21</td>
<td>29</td>
</tr>
<tr>
<td>R2D1</td>
<td>18</td>
<td>206 K</td>
<td>36 K</td>
<td>17</td>
<td>32</td>
</tr>
<tr>
<td>Esm02</td>
<td>17</td>
<td>307 K</td>
<td>44 K</td>
<td>14</td>
<td>32</td>
</tr>
<tr>
<td>Sat2</td>
<td>18</td>
<td>355 K</td>
<td>44 K</td>
<td>12</td>
<td>35</td>
</tr>
<tr>
<td>SL13</td>
<td>17</td>
<td>126 K</td>
<td>22 K</td>
<td>17</td>
<td>40</td>
</tr>
<tr>
<td>SL16</td>
<td>16</td>
<td>160 K</td>
<td>22 K</td>
<td>14</td>
<td>40</td>
</tr>
<tr>
<td>Esm01</td>
<td>18</td>
<td>222 K</td>
<td>30 K</td>
<td>14</td>
<td>48</td>
</tr>
<tr>
<td>SL14</td>
<td>14</td>
<td>215 K</td>
<td>43 K</td>
<td>20</td>
<td>52</td>
</tr>
<tr>
<td>SL15</td>
<td>17</td>
<td>330 K</td>
<td>43 K</td>
<td>13</td>
<td>56</td>
</tr>
</tbody>
</table>

Data sets are ranked by number of clusters.

J Neurophysiol • doi:10.1152/jn.01046.2012 • www.jn.org
The decoding performance is critically dependent on the choice of kernel bandwidths \( h_x \) and \( h_y \). Optimal bandwidths that minimized decoding error were selected separately for each decoder variant and each data set (Table 3; see METHODS). For the spike feature decoder, bandwidths clustered around \( h_x = 6 \) cm and \( h_y = 24 \) μV (Fig. 3). Decoding performance decreased gradually for increasing bandwidths. If the bandwidths were fixed at \( h_x = 6 \) cm and \( h_y = 24 \) μV for all data sets, the mean deviation from the median error obtained when optimized bandwidths were used was 5% (maximum: 17%). This suggests that even in cases where cross-validation is impractical, appropriate fixed bandwidth parameters selected a priori may still yield near-optimal decoding performance for distinct data sets.

### Comparison with Decoding Using Sorted Single Units

Next, we compared the spike feature decoding approach to the standard practice in which spikes recorded on tetrodes are first sorted into separate single units ("cluster tag decoder"; see Zhang et al. 1998). The automated sorting procedure produced a large number of putative clusters, of which 9–56 clusters across all data sets met the minimum quality criteria (Table 1), and the fraction of sorted spikes out of all putative pyramidal neuron spikes ranged from 6% to 21%. The remaining clusters on each tetrode were grouped into a separate hash cluster that was also included in the decoding. At the group level, the spike feature decoder resulted in a significantly lower median error than the cluster tag decoder (Fig. 4A; paired rank sum test, \( P = 0.0068 \)), with an average improvement of 14%.

We next applied the spike feature decoding approach separately to spikes in the well-isolated clusters and spikes in the hash and compared the performance to the equivalent cluster tag decoder (Fig. 4B). For both decoders, the performance significantly decreased when only the well-isolated clusters or only the hash clusters were used (paired rank sum test; spike feature decoder: clusters + hash vs. clusters only, \( P = 4.9 \times 10^{-4} \); clusters + hash vs. hash only, \( P = 4.9 \times 10^{-4} \); cluster tag decoder: clusters + hash vs. clusters only, \( P = 0.0024 \), clusters + hash vs. hash only, \( P = 4.9 \times 10^{-4} \)). When only the well-isolated clusters were used for decoding, the spike feature decoder showed a tendency toward a lower median error (average benefit: 7%; paired rank sum test: \( P = 0.043 \)).

### Table 2. **Summary of decoding performances across all data sets**

<table>
<thead>
<tr>
<th>Data Set</th>
<th>All spikes</th>
<th>Spikes in isolated clusters</th>
<th>Spikes in hash</th>
<th>MUA Decoder, All Spikes</th>
<th>Isolated clusters + hash</th>
<th>Isolated clusters</th>
<th>Hash only</th>
<th>All clusters</th>
</tr>
</thead>
<tbody>
<tr>
<td>RID2</td>
<td>6.0</td>
<td>28.0</td>
<td>18.5</td>
<td>23.0</td>
<td>11.0</td>
<td>36.3</td>
<td>31.3</td>
<td>7.5</td>
</tr>
<tr>
<td>R2D2</td>
<td>6.0</td>
<td>18.0</td>
<td>13.5</td>
<td>12.0</td>
<td>7.0</td>
<td>21.5</td>
<td>17.5</td>
<td>7.0</td>
</tr>
<tr>
<td>FK11</td>
<td>6.2</td>
<td>9.9</td>
<td>7.1</td>
<td>25.3</td>
<td>7.2</td>
<td>9.1</td>
<td>23.5</td>
<td>5.9</td>
</tr>
<tr>
<td>R1D1</td>
<td>4.5</td>
<td>7.5</td>
<td>7.0</td>
<td>13.5</td>
<td>5.5</td>
<td>8.5</td>
<td>24.0</td>
<td>4.5</td>
</tr>
<tr>
<td>R2D1</td>
<td>5.5</td>
<td>5.0</td>
<td>4.1</td>
<td>12.2</td>
<td>4.5</td>
<td>5.8</td>
<td>12.6</td>
<td>3.0</td>
</tr>
<tr>
<td>Exm02</td>
<td>3.6</td>
<td>5.0</td>
<td>6.2</td>
<td>12.7</td>
<td>6.9</td>
<td>7.1</td>
<td>16.1</td>
<td>5.9</td>
</tr>
<tr>
<td>Sat2</td>
<td>6.0</td>
<td>6.8</td>
<td>6.2</td>
<td>12.7</td>
<td>4.6</td>
<td>5.5</td>
<td>19.8</td>
<td>4.9</td>
</tr>
<tr>
<td>SL13</td>
<td>4.7</td>
<td>4.9</td>
<td>6.2</td>
<td>13.9</td>
<td>4.3</td>
<td>4.8</td>
<td>34.8</td>
<td>3.6</td>
</tr>
<tr>
<td>SL16</td>
<td>3.7</td>
<td>4.8</td>
<td>4.5</td>
<td>18.2</td>
<td>4.3</td>
<td>4.8</td>
<td>34.8</td>
<td>3.6</td>
</tr>
<tr>
<td>Exm01</td>
<td>6.0</td>
<td>6.2</td>
<td>6.3</td>
<td>9.0</td>
<td>5.3</td>
<td>5.8</td>
<td>9.9</td>
<td>5.0</td>
</tr>
<tr>
<td>SL14</td>
<td>3.6</td>
<td>4.1</td>
<td>4.3</td>
<td>13.9</td>
<td>4.0</td>
<td>4.1</td>
<td>26.3</td>
<td>3.5</td>
</tr>
<tr>
<td>SL15</td>
<td>3.2</td>
<td>3.8</td>
<td>4.8</td>
<td>63.2</td>
<td>4.2</td>
<td>3.9</td>
<td>102.6</td>
<td>3.6</td>
</tr>
</tbody>
</table>

Decoding performance is presented as median error in centimeters. Data sets are ranked by number of clusters. MUA, multiunit activity.
Innovative Methodology

BAYESIAN DECODING USING UNSORTED SPIKES

A

B

C

D

E

J Neurophysiol • doi:10.1152/jn.01046.2012 • www.jn.org
The cluster tag decoder may therefore benefit from using the original feature decoder is mainly due to the extraction of more information from the hash. This suggests that the improved performance of the spike feature decoder may be higher than the four-amplitude tetrode scenario (38% vs. 14%). This result suggests that the relative benefit of spike sorting, and we showed above that decoding performance decreases when fewer features are available (Fig. 2 E). Spike sorting quality depends on the number of (informative) spike waveform features. With fewer features, cluster isolation becomes poorer and hence decoding performance in the cluster tag decoder will decrease. The spike feature decoder taps into the same information in the spike waveform features as spike sorting, and we showed above that decoding performance decreases when fewer features are available (Fig. 2E).

Here we investigated the relative performance of the spike feature decoder versus the cluster tag decoder when fewer features from a single electrode are available. Spike sorting proceeded as for tetrodes and resulted in fewer well-isolated clusters (Table 4). In this scenario, the spike feature decoder significantly outperformed the cluster tag decoder (Fig. 4D; paired rank sum test $P = 0.0068$), with an average improvement that is higher than the four-amplitude tetrode scenario (38% vs. 14%). This result suggests that the relative benefit of using the spike feature decoder may be higher when fewer spike features are available for spike sorting.

**DISCUSSION**

We have demonstrated a novel Bayesian neural decoding approach that implements a direct mapping between spike waveform features and a sensory stimulus or other covariates.
This new approach has several important advantages over other methods. First, the spike feature decoder uses both the timing and waveform features of all available spike events to increase the amount of stimulus-related information that can be extracted from the neural signals. Second, there is no need for an intermediate spike sorting step that adds complexity and may lead to loss of information. Third, our approach defines a nonparametric encoding model to flexibly describe the relation between the stimulus and neural activity. And finally, the spike feature decoder allows both encoding and decoding stages to be performed with minimal supervision.

We tested the new decoding method on tetrode recordings from the hippocampus in freely behaving rats and showed that the animal’s location on a track could be accurately estimated. In our tests we used spike peak amplitude as the feature for decoding; however, it is possible to use any other set of spike features as well, for example, waveform width, wave shape parameters, or PCs. Our approach takes advantage of the same spike waveform information as most spike sorting methods but does so without an explicit sorting step. Our results show a small benefit of the spike feature decoder compared with a standard cluster tag decoder, as the new approach extracts more information from the poorly separable hash spikes.

It is noteworthy to point out the key differences between our approach and other paradigms in which neural spiking data were used for decoding without a spike sorting step. Fraser et al. (2009) fitted MUA responses to movement with a spline function, and thus each electrode was treated as a single “virtual” unit. This approach may work well if only a few neurons are recorded on each electrode, or if all neurons contributing to the MUA have similar tuning properties. However, this method will likely perform poorly if the MUA contains spikes from many neurons with diverse responses, for example, in the hippocampus as we showed here.

Stark and Abeles (2007) presented another interesting approach in which MUA, defined as the root mean square of the 300 Hz to 6 kHz band of the local field potential, was used to predict arm movement in monkeys. In this method, spiking activity was not modeled explicitly and decoding was performed by a support vector machine classifier. Unlike our approach, the measure of MUA in Stark and Abeles (2007) does not separate the contributions of spike rate and amplitude, nor does it allow incorporation of other waveform features.

Ventura (2008) proposed a paradigm in which the identities of neurons are extracted implicitly from the spike train recorded on a single electrode by assuming a known parametric model for the neurons’ tuning to the stimulus. This approach worked well in simulations but was not applied to a real data set, and it is not clear that the method will work equally well for populations of neurons with complex, nonparametric tuning functions. In addition, the decoding step assumes that the temporal evolution of the stimulus is smooth (Ventura 2008). This assumption is not necessarily true or known to be true, for example, if the goal is to decode a hidden stimulus with unknown temporal dynamics (Davidson et al. 2009; Kloosterman 2012).

More recently, Ventura proposed to exploit the information in covariates that modulate neuronal firing (“tuning informa-

Fig. 3. Optimal bandwidths \( h_x \) and \( h_a \) for spike feature decoder for all data sets. The mean optimal bandwidth is indicated by the “+” symbol. Dashed grid lines indicate bandwidth values used for cross-validation.

Fig. 4. Comparison of the spike feature decoder to the standard single-unit decoder for 3 different decoding strategies. In all panels, box plots compare the median error distributions for the 2 decoders. Gray lines represent data of all individual data sets. Outliers are indicated by the “+” symbol. A: decoding using all spikes in isolated clusters and hash. B: decoding using only spikes in isolated clusters, excluding hash. C: decoding using only spikes in hash. D: comparison of spike feature decoder and cluster tag decoder using a reduced set of spike waveform features from a single electrode.
tion”) in addition to spike waveform information to improve spike sorting (Ventura 2009a) with application in decoding as well (Ventura 2009b). Ventura’s approach uses a mapping between waveform features and covariates similar to what we have done in the spike feature decoder; however, parametric models are used rather than the nonparametric approach in our work.

Luczak and Narayanan (2005) avoided spike sorting by constructing a “spectral representation” of the neural data by considering a discrete space of spike waveform features (they used PCs) and time. The spectral representation of Luczak and Narayanan is similar to the spike feature-marked point process that is at the basis of our method. Using a combination of partial least squares to select stimulus-relevant features and linear discriminant analysis for classification, they demonstrated that the spectral representation can be used to successfully discriminate between two discrete auditory stimuli. In contrast, we used a naive Bayesian approach in which spiking statistics are explicitly modeled, and which we applied to the decoding of a continuously time-varying stimulus.

The main strength of the proposed spike feature decoding approach is that it provides straightforward stimulus estimation from information-rich spiking data in a minimally supervised manner. The method can be applied in online and real-time encoding/decoding scenarios, which makes it appealing for brain-machine interfaces and neural prosthetics that use chronic implants. By utilizing the information carried by all spikes, whether or not they can be uniquely assigned to a single neuron, the spike feature decoding approach is more robust to changes in signal quality as observed in chronic recording applications. In addition, nonstationarity of the neural signals, which is commonly encountered in long-term recordings, and slow changes in the encoding model can be easily handled by restricting the spike events that contribute to the encoding model to a finite temporal window.

The spike feature decoding approach is not fully unsupervised, as it requires the selection of appropriate kernel bandwidth parameters. In the present experimental data analysis, a cross-validation approach was employed to estimate the two bandwidth parameters for spike amplitude and position dimensions. We found that the decoding performance was relatively

<table>
<thead>
<tr>
<th>Data Set</th>
<th>No. of Electrodes</th>
<th>No. of Total Spikes</th>
<th>No. of sorted spikes</th>
<th>% Sorted spikes</th>
<th>No. of clusters</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1D2</td>
<td>14</td>
<td>55 K</td>
<td>36 K</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>R2D2</td>
<td>18</td>
<td>77 K</td>
<td>53 K</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>FK11</td>
<td>9</td>
<td>146 K</td>
<td>104 K</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>R1D1</td>
<td>14</td>
<td>132 K</td>
<td>102 K</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>R2D1</td>
<td>18</td>
<td>158 K</td>
<td>121 K</td>
<td>10</td>
<td>6</td>
</tr>
<tr>
<td>Sat2</td>
<td>18</td>
<td>290 K</td>
<td>218 K</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>SL13</td>
<td>17</td>
<td>111 K</td>
<td>68 K</td>
<td>11</td>
<td>6</td>
</tr>
<tr>
<td>SL16</td>
<td>16</td>
<td>127 K</td>
<td>75 K</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>Esm01</td>
<td>18</td>
<td>170 K</td>
<td>121 K</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>SL14</td>
<td>14</td>
<td>180 K</td>
<td>110 K</td>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td>SL15</td>
<td>17</td>
<td>280 K</td>
<td>164 K</td>
<td>11</td>
<td>6</td>
</tr>
</tbody>
</table>

Data sets are ranked as in Table 1.

Table 4. Summary of experimental data for single-electrode and stereotrode simulation

Fig. 5. Simulated minimally supervised online decoding. A: posterior probability distributions for 250-ms time bins in 3 epochs corresponding to the first 2 laps and the last lap on the track (data set SL14). Darker shades of gray indicate higher probabilities. Solid line is animal’s true trajectory. B: median decoding error computed for each lap separately. Horizontal dashed line shows mean of the median decoding error statistics across all laps.
Innovative Methodology

Bayesian Decoding Using Unsorted Spikes

Robust around suboptimal bandwidth parameters, suggesting that a priori selection of a fixed bandwidth may be appropriate. For online applications it is important that the computations for encoding and decoding stages can be performed in real time. In our present implementation encoding is cheap (all recorded spikes are simply retained), whereas decoding is computationally intensive. The complexity of decoding scales with the total number of spike events incorporated in the encoding model. Several strategies can be used to decrease the computational burden. If precomputation of the rate function is practical, then this would avoid costly evaluation of the densities during decoding. If necessary, a trade-off can be made between decoding performance and computation speed by reducing the number of dimensions of the feature space. For example, using two amplitude features (from a stereotrode) instead of four features (from a tetrode) gives acceptable (albeit decreased) decoding performance. Alternatively, it is possible to reduce computational load of online density evaluation by excluding groups of low-information spike events based on their waveform properties. For example, in the hippocampus most stimulus-relevant information is carried by pyramidal neuron activity, and therefore in our tests we filtered out spikes from putative interneurons (peak-trough latency < 0.35 ms). This selection of spikes substantially reduced the computational burden without significantly affecting decoding performance. A further reduction can be achieved by increasing the spike amplitude threshold, excluding low-amplitude spikes that carry relatively little information. Another way to alleviate this problem is to find compact and efficient representations of the kernel density estimates of the rate functions (Girolami 2003; Huang and Chow 2006; Mitra et al. 2002; Zhou et al. 2003).

Our decoding algorithm is based on the statistical assumption that the spike events follow a spatio-temporal Poisson process (i.e., spike events are mutually independent in both spike feature space and time). This is equivalent to assuming an independent Poisson rate code for all neurons in the ensemble. In our example, the encoding model is constructed from multiple presentations of the same stimulus, and even if individual spike trains do not follow a Poisson distribution, the aggregate of many such spike trains will. Although the Poisson assumption is oversimplified for most experimental data, it provides us with a simple and tractable solution for decoding analysis. It is possible to relax the assumptions of Poisson statistics and independence—for example, to incorporate temporal dependence the spike history in each electrode can be represented as an augmented temporal feature. Alternatively, the spatial local dependence can be modeled by considering a Neyman-Scott process (Diggle 2003) or a Poisson cluster process (Bartlett 1964; Wolpert and Ickstadt 1998). These topics will be the subject of future investigations.

Our decoding paradigm can be extended in several ways. For instance, we can reformulate the Bayesian decoding problem within the state-space framework by inclusion of a smooth temporal prior (Brown et al. 1998). A temporal prior can be useful for reconstruction of stimuli or covariates that are known to change smoothly over time, such as the position of a rat or the movement of an arm. Spike feature decoding can also be used to extract information from ensemble spiking activity by estimating the mutual information and entropy of the stimulus and neural responses. Estimation of these information measures may provide a better understanding of the underlying principles of neural codes used in the brain (Jacobs et al. 2009; Quijan Quiroga and Panzeri 2009).

Acknowledgments

We thank Greg Hale and Jun Yamamoto for contributing their data sets.

Grants

This work was supported by National Institute of Mental Health Grant MH-061976 and Office of Naval Research MURI N00014-10-1-0936 Grant to M. A. Wilson. Z. Chen was supported by an Early Career Award from the Mathematical Biosciences Institute (MBI) and a National Science Foundation-CRCNS (Collaborative Research in Computational Neuroscience) grant.

Disclosures

No conflicts of interest, financial or otherwise, are declared by the author(s).

Author Contributions


References


J Neurophysiol • doi:10.1152/jn.01046.2012 • www.jn.org


