Electrophysiological Spike Sorting via Joint Dictionary Learning & Mixture Modeling

Qisong Wu, David E. Carlson, Wenzhao Lian, Mingyuan Zhou, Colin R. Stoetzner, Daryl Kipke, Douglas Weber, Joshua T. Vogelstein, David B. Dunson and Lawrence Carin

***Abstract*—A new model is developed for feature learning and clustering of electrophysiological data across multiple recording periods. The model is applicable to situations in which the detected spikes may be clipped (constituting missing data). [c3]We demonstrate that joint feature (dictionary) learning and [c4]mixture modeling [c5](clustering) allows one to [c6] [c7]distinguish single-unit spikes from non-local phenomena and artifacts[c8]. We explicitly model the number of spikes within a measurement interval, addressing a time-evolving firing rate. Further, we model the number of clusters, mitigating limitations of methods like the Dirichlet process. Model properties are discussed, state-of-the-art results are presented on public data, and the methodology is demonstrated on new [c9] experimental [c10] data.**

***Index Terms*—spike sorting, Bayesian, clustering, Dirichlet process**

I. Introduction

BRAIN-MACHINE interfaces often utilize a sensor array to measure [c11]electrophysiological[c12] activity within regions of the brain, with the ultimate goal of controlling robotic limbs [1] or muscles. When processing electrical signals from such a device, one typically (i) filters the raw sensor readings, (ii) performs thresholding to "detect" the spikes, (iii) maps each detected spike to a feature vector, and (iv) then clusters the feature vectors [15]. The complexities of real data[c13] from awake/moving animals[c14] may significantly complicate the characteristics of the data[c15]. That the spike detection phase is imperfect[c16] motivates reconsideration of aspects of this analysis chain.

Q. Wu, D. Carlson, W. Lian, M. Zhou and L. Carin are with the Department of Electrical and Computer Engineering, Duke University, Durham, NC, USA

C.R. Stoetzner and D. Kipke are with the Department of Biomedical Engineering, University of Michigan, Ann Arbor, MI, USA

D. Weber is with the Department of Biomedical Engineering, University of Pittsburgh, Pittsburgh, PA, USA

J. Vogelstein and D. Dunson are with the Department of Statistical Science, Duke University, Durham, NC, USA

Manuscript received October 27, 2012.

[c3] ~~It is demonstrated~~

[c4] ~~clustering~~

[c5] *Text added.*

[c6] ~~perform forensics on the characteristics of the data (distinguishing~~

[c7] *Text added.*

[c8] ~~)~~

[c9] ~~measured~~

[c10] ~~ephys~~

[c11] ~~electrical (~~

[c12] ~~, or "ephys")~~

[c13] ~~,~~

[c14] ~~,~~

[c15] ~~, and~~

[c16] ~~, warranting~~

Concerning the detection phase, not all signals that exceed a threshold are neuronal spikes (localized wavelet-like signals, henceforth also termed "single unit events"). For example, there are biologically significant signals, such as local field potentials, that may not be spikes [7]. Some signals that exceed the threshold may be due to artifacts; these may be manifested as indirect effects of movement on the recording apparatus, for example when a behaving animal collides with objects, grooms near the implant site, or chews. These behaviors interfere with the electromechanical interface between the headstage (attached to the recording cable) and implant assembly (anchored to dental acrylic on the animal's head). In addition, biological signals that occur near the reference electrode may also exceed the threshold. There is therefore a need for additional steps to distinguish spike-like and non-spike-like signals after step (ii), and to identify artifact signals.

Concerning feature extraction, step (iii), this is typically [c17]implemented prior to subsequent clustering, with principal components analysis (PCA) [15] and wavelets [14] representing popular methods. [c18]PCA is problematic as it requires choosing or tuning a hyperparameter specifying the number of components to keep. Moreover, neither PCA nor wavelets are robust to physiological noise[c19c20] [**?**]. To infer the number of clusters based upon the observed data, mixture models have become increasingly popular, and nonparametric Bayesian methods have proven effective [6], [9]. Researchers have also recently employed mixture of factor analyzers to *jointly* perform feature extraction and clustering in a data-adaptive manner [6], [10], combining steps (iii) and (iv) above.

A practical issue that has received limited attention concerns another imperfection in the detection phase: automatic detection algorithms that extract spikes and then discard the raw data (yielding desired data compression) may be imperfect, and part of a spike signal may be clipped off, and discarded. If this occurs, traditional feature learning algorithms like PCA or wavelets [c21]often fail. The dictionary-learning-based framework developed here[c22]—by virtue of employing an explicit

[c17] ~~done~~

[c18] ~~For PCA one must *a priori* select the number of principal components (or employ trial and error to determine the proper/appropriate number of components);~~

[c19] ~~while wavelets may be spike-like, they were not designed to be matched to electrophysiological data (and as indicated above, not all signals that pass a threshold are spike-like)~~

[c20] i don't actually see how not being designed for something is a bad thing. it is bad if it doesn't work. but if the tool fits....

[c21] ~~cannot be employed~~

[c22] *Text added.*

generative model—may be used to impute the missing data, and therefore perform feature extraction even when the spike extraction is imperfect.

The use of nonparametric Bayesian methods like the Dirichlet process (DP) [6], [9] removes some of the *ad hoc* character of classical clustering methods, but there are other limitations within the context of electrophysiological data analysis. The DP and related models are characterized by a scale parameter $\alpha > 0$, and the number of clusters grows as $\mathcal{O}(\alpha \log S)$ [17], with S the number of data samples. This growth without limit in the number of clusters with increasing data is undesirable in the context of electrophysiological data, for which there are a finite set of processes responsible for the observed data. Further, when jointly performing mixture modeling across multiple tasks, the *hierarchical* Dirichlet process (HDP) [18] shares all mixture components, which may undermine inference of subtly different clusters.

[c1] [c2] [c3] Another limitation of almost all existing electrophysiological data methods is that they operate on each putative spike independently. In contrast, we explicitly model the spike rate of each neuron (jointly with the clustering model), thereby enabling us to borrow strength across the collection of all spiking events.

In this paper we integrate dictionary learning and clustering for analysis of electrophysiological data, as in [6], [10]. However, as an alternative to utilizing a method like DP or HDP [6], [9] for clustering, we develop a new hierarchical clustering model in which the number of clusters is modeled explicitly; this implies that we model the number of underlying [c4] neurons—or clusters—separately from the firing rate, with the latter controlling the total number of observations. This is done by integrating the Indian buffet process (IBP) [11] with the Dirichlet distribution, similar to [21], but with unique characteristics. The IBP is a model that may be used to *learn* features representative of data, and each potential feature is a "dish" at a "buffet"; each data sample (here [c5] a neuronal spike) selects which features from the "buffet" are most appropriate for its representation. The Dirichlet distribution is used for clustering data, and therefore here we jointly perform feature learning and clustering, by integrating the IBP with the Dirichlet distribution. The proposed framework explicitly models the quantity of data ([c6] for example, spikes) measured within a given recording interval. We believe that this is the first time the firing rate of electrophysiological data is modeled jointly with clustering (and, here, jointly with feature/dictionary learning). The model demonstrates state-of-the-art clustering performance on publicly available data. Further, concerning distinguishing single-unit-events, we demonstrate how this may be achieved using the proposed method, considering new measured (experimental) electrophysiological data.

The remainder of the paper is organized as follows. In Section II we introduce the basic modeling framework, and make connections to previous related research. Methods by which computations are performed are discussed in Section III. Several sets of experimental results are presented in Section IV, followed in Section V by conclusions.

II. Models and Analysis

A. Bayesian dictionary learning

Consider electrophysiological data measured over a prescribed time interval. Specifically, let $\mathbf{X}_{ij} \in \mathbb{R}^{T \times N}$ represent the j[c7] th signal observed during interval i[c8], and let J_i be the number of detected waveforms for interval i. (Note that we could write j_i for clarity, but we drop the subscript i for brevity.). The data are assumed recorded on each of N channels, from an N-element sensor array, and there are T time points associated with each detected spike waveform (the signals are aligned in time with respect to their peak value). In tetrode arrays [8], and related devices like those considered below, a single-unit event ([c9] action potential of a neuron) may be recorded on multiple adjacent channels, and therefore it is of interest to process the N signals associated with $\mathbf{X}_{ij}$ jointly; the joint analysis of all N signals is also useful for data forensics, discussed in Section IV.

To constitute data $\mathbf{X}_{ij}$, [c10] we assume that threshold-based detection (or a related method) is performed on data measured from each of the N sensor channels. When a signal is detected on any of the channels, coincident data are also extracted from all N channels, within a window of (discretized) length T [c11] centered at the detection peak. On some of the channels data may be associated with a single-unit event, and on other channels the data may represent background noise. Both types of data (signal and noise) are modeled jointly, as discussed below.

Following [6], we employ dictionary learning to model each $\mathbf{X}_{ij}$; however, unlike [6] we jointly employ dictionary learning to all N channels in $\mathbf{X}_{ij}$ (rather than separately to each of the channels). The data are represented

$$\mathbf{X}_{ij} = \mathbf{D}\mathbf{\Lambda}\mathbf{S}_{ij} + \mathbf{E}_{ij}, \tag{1}$$

where $\mathbf{D} \in \mathbb{R}^{T \times K}$ represents a dictionary with K dictionary elements (columns), $\mathbf{\Lambda} \in \mathbb{R}^{K \times K}$ is a diagonal matrix with sparse diagonal elements, $\mathbf{S}_{ij} \in \mathbb{R}^{K \times N}$ represents the dictionary weights (factor scores), and $\mathbf{E}_{ij} \in \mathbb{R}^{T \times N}$ represents residual/noise. Let $\mathbf{D} = (\boldsymbol{d}_1, \ldots, \boldsymbol{d}_K)$ and $\mathbf{E} = (\boldsymbol{e}_1, \ldots, \boldsymbol{e}_N)$, with [c12] $\boldsymbol{d}_k, \boldsymbol{e}_n \in \mathbb{R}^T$. We impose [c13] priors

$$\boldsymbol{d}_k \sim \mathcal{N}(0, \frac{1}{T}\mathbf{I}_T) \ , \quad \boldsymbol{e}_n \sim \mathcal{N}(0, \text{diag}(\eta_1^{-1}, \ldots, \eta_T^{-1})), \tag{2}$$

[c1] ~~Another limitation of almost all existing electrophysiological data methods is that they only focus on clustering the observed data. While assigning data to a cluster is important, such frameworks do not address one of the most significant aspects of spike data: recent research indicates that a major portion of the information content related to neural spiking is carried in the spike *rate*, in terms of the number of spikes within a defined interval. It is therefore not only desirable to model the clustering of the data, but also the rate of spike firing, ideally with these modeled jointly.~~

[c2] also removed Donoghue07 citation

[c3] *Text added.*

[c4] ~~neural processes~~

[c5] *Text added.*

[c6] ~~*e.g.*~~

[c7] ~~th~~

[c8] *Text added.*

[c9] ~~*e.g.*~~

[c10] ~~it is assumed~~

[c11] *Text added.*

[c12] ~~$\boldsymbol{d}_k \in \mathbb{R}^T$ and $\boldsymbol{e}_n \in \mathbb{R}^T$~~

[c13] ~~the~~

where $\mathbf{I}_T$ is the $T \times T$ dimensional identity matrix [c14]and $\eta_t \in \mathbb{R}$ for all t.

We wish to impose that each column of $\mathbf{X}_{ij}$ lives in a linear subspace, with dimension and composition to be inferred. The composition of the subspace is defined by a selected subset of the columns of $\mathbf{D}$, and that subset is defined by the non-zero elements in the diagonal of $\mathbf{\Lambda} = \text{diag}(\boldsymbol{\lambda})$, with $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_K)^T$ [c1]and $\lambda_k \in \mathbb{R}$ for all k. We impose $\lambda_k \sim \nu\delta_0 + (1-\nu)\mathcal{N}_+(0, \alpha_0^{-1})$, with $\nu \sim \text{Beta}(a_0, b_0)$ and δ_0 a unit measure concentrated at zero. The hyperparameters [c2]$a_0, b_0 \in \mathbb{R}$ are set to encourage sparse $\boldsymbol{\lambda}$, and $\mathcal{N}_+(\cdot)$ represents a normal distribution truncated to be non-negative. Diffuse gamma priors are placed on $\{\eta_t\}$ and α_0.

Concerning the model priors, the assumption $\boldsymbol{d}_k \sim \mathcal{N}(0, \frac{1}{T}\mathbf{I}_T)$ is consistent with a conventional ℓ_2 regularization ([c3]shrinkage[c4]) on the dictionary elements. Similarly, the assumption $\boldsymbol{e}_n \sim \mathcal{N}(0, \text{diag}(\eta_1^{-1}, \ldots, \eta_T^{-1}))$ corresponds to an ℓ_2 fit of the data to the model, with a weighting on the norm as a function of the sample point (in time) of the signal. These priors are typically employed in dictionary learning; see [23] for a discussion of the connection between such priors and optimization-based dictionary learning.

B. Mixture modeling

A mixture model is imposed for the dictionary weights $\mathbf{S}_{ij} = (\boldsymbol{s}_{ij1}, \ldots, \boldsymbol{s}_{ijN})$, with $\boldsymbol{s}_{ijn} \in \mathbb{R}^K$; $\boldsymbol{s}_{ijn}$ [c5]defines the weights on the dictionary elements for the data associated with the nth channel (nth column) in $\mathbf{X}_{ij}$. Specifically,

$$\boldsymbol{s}_{ijn} \sim \mathcal{N}(\boldsymbol{\mu}_{z_{ij}n}, \mathbf{\Omega}_{z_{ij}n}^{-1}), \tag{3}$$

$$z_{ij} \sim \textstyle\sum_{m=1}^M \pi_m^{(i)} \delta_m, \;\; (\boldsymbol{\mu}_{mn}, \mathbf{\Omega}_{mn}) \sim G_0 \tag{4}$$

where G_0 is a normal-Wishart distribution, $\pi_m^{(i)} > 0$, $\sum_{m=1}^M \pi_m^{(i)} = 1$, and $\{\boldsymbol{s}_{ijn}\}_{n=1,N}$ are all associated with cluster z_{ij}; $z_{ij} \in \{1, \ldots, M\}$ is an indicator variable defining with which cluster $\mathbf{X}_{ij}$ is associated[c6], and M is a user-specified upper bound on the total number of clusters possible.

The use of the Gaussian model in (3) is convenient, as it simplifies computational inference, and the normal-Wishart distribution G_0 is selected because it is the conjugate prior for a normal distribution. The key novelty we wish to address in this paper concerns design of the mixture probability vector $\boldsymbol{\pi}^{(i)} = (\pi_1^{(i)}, \ldots, \pi_M^{(i)})^T$.

The vector $\boldsymbol{\pi}^{(i)}$ defines the probability with which each of the M mixture components are employed for data recording interval i. We wish to place a prior probability distribution on $\boldsymbol{\pi}^{(i)}$, and to infer an associated posterior distribution based upon the observed data. A typical prior for $\boldsymbol{\pi}^{(i)}$ is a symmetric Dirichlet distribution [10],

$$\boldsymbol{\pi}^{(i)} \sim \text{Dir}(\tilde{\alpha}_0/M, \ldots, \tilde{\alpha}_0/M). \tag{5}$$

In the limit[c7], $M \to \infty$[c8], this reduces to a draw from a Dirichlet process [6], [9], represented $\boldsymbol{\pi}^{(i)} \sim \text{DP}(\tilde{\alpha}_0 G_0)$, with G_0 the "base" distribution defined in (4). Rather than drawing each $\boldsymbol{\pi}^{(i)}$ independently [c9]from DP($\tilde{\alpha}_0 G_0$), we may consider the hierarchical Dirichlet process (HDP) [18] as

$$\boldsymbol{\pi}^{(i)} \sim \text{DP}(\tilde{\alpha}_1 G) \;, \qquad G \sim \text{DP}(\tilde{\alpha}_0 G_0) \tag{6}$$

The HDP construction imposes that the $\{\boldsymbol{\pi}^{(i)}\}$ share the same set of "atoms" $\{\boldsymbol{\mu}_{mn}, \mathbf{\Omega}_{mn}\}$, implying a sharing of the different types of clusters across the time intervals i at which data are collected. A detailed discussion of the HDP formulation is provided in [6].

These models have limitations in that the inferred number of clusters grows with observed data (here the clusters are ideally connected to [c10]neurons, the number of which will not necessarily grow with [c11]longer samples). Further, the above clustering model assumes the number of samples is given, and hence is not modeled (the information-rich firing rate is not modeled). Below we develop a framework that yields hierarchical clustering like HDP, but the number of clusters and the data count ([c12]for example, spike rate) are modeled explicitly.

C. Hierarchical count and mixture modeling

Let the total set of data measured during interval i be represented $\boldsymbol{\mathcal{D}}_i = \{\mathbf{X}_{ij}\}_{j=1}^J$ [c13] . In the experiments below, a "recording interval" corresponds to a day on which data were recorded for an hour (data are collected separately on a sequence of days), and the set $\{\mathbf{X}_{ij}\}_{j=1,M_i}$ defines all signals that exceeded a threshold during that recording period. In addition to modeling M_i, we wish to infer the number of distinct clusters C_i characteristic of $\boldsymbol{\mathcal{D}}_i$, and the relative fraction (probability) with which the M_i observations are apportioned to the C_i clusters.

Let n_{im}^* represent the number of data samples in $\boldsymbol{\mathcal{D}}_i$ that are apportioned to cluster $m \in \{1, \ldots, M\} = \mathcal{S}$, with $M_i = \sum_{m=1}^M n_{im}^*$. The set $\mathcal{S}_i \subset \mathcal{S}$, with $C_i = |\mathcal{S}_i|$, defines the *active* set of clusters for representation of $\boldsymbol{\mathcal{D}}_i$, and therefore M serves as an upper bound ($n_{im}^* = 0$ for $m \in \mathcal{S} \setminus \mathcal{S}_i$).

We impose $n_{im}^* \sim \text{Poisson}(b_m^{(i)} \hat{\phi}_m^{(i)})$ with

$$\hat{\phi}_m^{(i)} \sim \text{Ga}(\phi_m, p_i/(1-p_i)) \tag{7}$$

$$b_m^{(i)} \sim \text{Bern}(\nu_m), \;\; p_i \sim \text{Beta}(a_0, b_0), \;\; \phi_m \sim \text{Ga}(\gamma_0, 1) \tag{8}$$

$$\nu_m \sim \text{Beta}(\alpha/M, 1), \;\; \gamma_0 \sim \text{Ga}(c_0, 1/d_0) \tag{9}$$

where $\text{Ga}(\cdot)$ denotes the gamma distribution, and $\text{Bern}(\cdot)$ the Bernoulli distribution. Note that $\{\phi_m, \nu_m\}_{m=1,M}$ are shared across all intervals i, and it is in this manner we achieve joint

[c14] *Text added.*
[c1] *Text added.*
[c2] ~~(a_0, b_0)~~
[c3] ~~imposition of smoothness~~
[c4] my sense is that ℓ_2 does not smooth, ℓ_2 of the gradient smoothes, a simple ℓ_2 just shrinks.
[c5] ~~refines~~
[c6] *Text added.*
[c7] *Text added.*
[c8] *Text added.*
[c9] ~~$\boldsymbol{\pi}^{(i)} \sim \text{DP}(\tilde{\alpha}_0 G_0)$~~
[c10] ~~neural processes~~
[c11] ~~increasing data~~
[c12] ~~*e.g.*~~
[c13] i replaced M_i here with J, because it is $z_{ij} \in \{1, \ldots, M_i\}$, not j; rather, j indexes which spike event we are talking about, and z_{ij} indicates which unit does spike event (i, j) corresponds to. again, please let me know if i am wrong, i have replaced the other M_i's with J where i thought appropriate.

clustering across all intervals, like via HDP. The reasons for the choices of these various priors is discussed in Section II-D, when making connections to related models. For example, the choice $b_m^{(i)} \sim \text{Bern}(\nu_m)$ with $\nu_m \sim \text{Beta}(\alpha/M, 1)$ is motivated by the connection to the Indian buffet process [11] as $M \rightarrow \infty$.

Note that we explicitly model the number of clusters and quantity of data within a given cluster. This implies that data are mapped to the same cluster if they have consistent signal shape *and* if the associated firing rate is consistent (note that the firing rates for some individual neurons can vary widely – from a few spikes/sec to 100+; motor neuron firing rates are typically much lower and less variable). Consequently both the signal shape and firing rate dictates how the data are clustered.

Note that $n_{im}^* = 0$ when $b_m^{(i)} = 0$, and therefore $\boldsymbol{b}^{(i)} = (b_1^{(i)}, \ldots, b_M^{(i)})^T$ defines indicator variables identifying the active subset of clusters $\mathcal{S}_i$ for representation of $\boldsymbol{\mathcal{D}}_i$. Marginalizing out $\hat{\phi}_m^{(i)}$, $n_{im}^* \sim \text{NegBin}(b_m^{(i)}\phi_m, p_i)$. This emphasize another motivation for the form of the prior: the negative binomial modeling of the counts (firing rate) is more flexible than a Poisson model, as it allows the mean and variance on the number of counts to be different (they are the same for a Poisson model).

While the above construction yields a generative process for the number, n_{im}^*, of elements of $\boldsymbol{\mathcal{D}}_i$ apportioned to cluster m, it is desirable to explicitly associate each member of $\boldsymbol{\mathcal{D}}_i$ with one of the clusters (to know not just *how many* members of $\boldsymbol{\mathcal{D}}_i$ are apportioned to a given cluster, but also *which* data are associated with a given cluster). Toward this end, consider the alternative equivalent generative process for $\{n_{im}^*\}_{m=1,M}$ (see Lemma 4.1 in [24] for a proof of equivalence): first draw $M_i \sim \text{Poisson}(\sum_{m=1}^M b_m^{(i)}\hat{\phi}_m^{(i)})$, and then

$$(n_{i1}^*, \ldots, n_{iM}^*) \sim \text{Mult}(M_i; \pi_1^{(i)}, \ldots, \pi_M^{(i)}) \quad (10)$$

$$\pi_m^{(i)} = b_m^{(i)}\hat{\phi}_m^{(i)} / \sum_{m'=1}^M b_{m'}^{(i)}\hat{\phi}_{m'}^{(i)} \quad (11)$$

with $\hat{\phi}_m^{(i)}$, $\{\phi_m\}$, $\{b_m^{(i)}\}$, and $\{p_i\}$ constituted as in (7)-(9). Note that we have $M_i \sim \text{NegBin}(\sum_{m=1}^M b_m^{(i)}\phi_m, p_i)$ by marginalizing out $\hat{\phi}_m^{(i)}$.

Rather than drawing $(n_{i1}^*, \ldots, n_{iM}^*) \sim \text{Mult}(M_i; \pi_1^{(i)}, \ldots, \pi_M^{(i)})$, for each of the M_i data we may draw indicator variables $z_{ij} \sim \sum_{m=1}^M \pi_m^{(i)}\delta_m$, where δ_m is a unit measure concentrated at the point m. Variable z_{ij} assigns data sample $j \in \{1, \ldots, M_i\}$ to one of the M possible clusters, and $n_{im}^* = \sum_{j=1}^{M_i} 1(z_{ij} = m)$, with $1(\cdot)$ equal to one if the argument is true, and zero otherwise. The probability vector $\boldsymbol{\pi}^{(i)}$ defined in (11) is now used within the mixture model in (4).

In the context of modeling and analyzing electrophysiological data, recent work on clustering models has accounted for refractory-time violations [6], [9], which occur when two or more spikes that are sufficiently proximate are improperly associated with the same cluster/neuron (which is impossible physiologically due to the refractory time delay required for the same neuron to re-emit a spike). The methods developed in [6], [9] may be extended to the class of mixture models developed above. We have not done so for two reasons: (i) in the context of everything else that is modeled here (joint feature learning, clustering, and count modeling), the refractory-time-delay issue is a relatively minor issue in practice; and (ii) perhaps more importantly, an important issue is that not all components of electrophysiological data are spike related (which are associated with refractory-time issues). As demonstrated in Section IV, a key component of the proposed method is that it allows us to distinguish single-unit (spike) events from other phenomena.

D. Relationship to existing models

As a consequence of the manner in which $\hat{\phi}_m^{(i)}$ is drawn in (7), and the definition of $\boldsymbol{\pi}^{(i)}$ in (11), for *any* $p_i \in (0, 1)$, the proposed model imposes

$$\boldsymbol{\pi}^{(i)} \sim \text{Dir}(b_1^{(i)}\phi_1, \ldots, b_M^{(i)}\phi_M) \quad (12)$$

Hence the proposed model is a generalization of (5). Considering the limit $M \rightarrow \infty$, and upon marginalizing out the $\{\nu_m\}$, the binary vectors $\{\boldsymbol{b}^{(i)}\}$ are drawn from the Indian buffet process (IBP), denoted $\boldsymbol{b}^{(i)} \sim \text{IBP}(\alpha)$. The number of non-zero components in each $\boldsymbol{b}^{(i)}$ is drawn from Poisson(α), and therefore for finite α the number of non-zero components in $\boldsymbol{b}^{(i)}$ is finite, even when $M \rightarrow \infty$. Consequently $\text{Dir}(b_1^{(i)}\phi_1, \ldots, b_M^{(i)}\phi_M)$ is well defined even when $M \rightarrow \infty$ since, with probability one, there are only a finite number of non-zero parameters in $(b_1^{(i)}\phi_1, \ldots, b_M^{(i)}\phi_M)$. This model is closely related to the compound IBP Dirichlet (CID) process developed in [21], with the following differences.

Above we have explicitly derived the relationship between the negative binomial distribution and the CID, and with this understanding we recognize the importance of p_i; the CID *assumes* $p_i = 1/2$, but there is no theoretical justification for this. Note that $M_i \sim \text{NegBin}(\sum_{m=1}^M b_m^{(i)}\phi_m^{(i)}, p_i)$. The mean of M_i is $(\sum_{m=1}^M b_m^{(i)}\phi_m)p_i/(1-p_i)$, and the variance is $(\sum_{m=1}^M b_m^{(i)}\phi_m)p_i/(1-p_i)^2$. If p_i is fixed to be 0.5 as in [21], this implies that we believe that the variance is two times the mean, and the mean and variance of M_i are the same for all intervals i and i' for which $\boldsymbol{b}^{(i)} = \boldsymbol{b}^{(i')}$. However, in the context of electrophysiological data, the rate at which neurons fire plays an important role in information content [7]. Therefore, there are many cases for which intervals i and i' may be characterized by firing of the same neurons (*i.e.*, $\boldsymbol{b}^{(i)} = \boldsymbol{b}^{(i')}$) but with very different rates ($M_i \neq M_{i'}$). The modeling flexibility imposed by inferring p_i therefore plays an important practical role for modeling electrophysiological data, and likely for other clustering problems of this type.

To make a connection between the proposed model and the HDP, motivated by (7)-(9), consider $\bar{\boldsymbol{\phi}} = (\bar{\phi}_1, \cdots, \bar{\phi}_M) \sim \text{Dir}(\gamma_0, \cdots, \gamma_0)$, which corresponds to $(\phi_1, \ldots, \phi_M)/\sum_{m'=1}^M \phi_{m'}$. From $\bar{\boldsymbol{\phi}}$ we yield a *normalized* form of the vector $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_M)$. The normalization constant $\sum_{m=1}^M \phi_m$ is lost after drawing $\bar{\boldsymbol{\phi}}$; however, because $\phi_m \sim \text{Ga}(\gamma_0, 1)$, we may consider drawing $\tilde{\alpha}_1 \sim \text{Ga}(M\gamma_0, 1)$, and *approximating* $\boldsymbol{\phi} \approx \tilde{\alpha}_1\bar{\boldsymbol{\phi}}$. With this approximation for $\boldsymbol{\phi}$, $\boldsymbol{\pi}^{(i)}$ may be drawn approximately as $\boldsymbol{\pi}^{(i)} \sim \text{Dir}(\tilde{\alpha}_1 b_1^{(i)}\bar{\phi}_1, \ldots, \tilde{\alpha}_1 b_M^{(i)}\bar{\phi}_M)$. This yields a simplified

and approximate hierarchy

$$\boldsymbol{\pi}^{(i)} \sim \text{Dir}(\tilde{\alpha}_1(\boldsymbol{b}^{(i)} \odot \bar{\boldsymbol{\phi}})) \quad (13)$$
$$\bar{\boldsymbol{\phi}} = (\bar{\phi}_1, \cdots, \bar{\phi}_M) \sim \text{Dir}(\gamma_0, \cdots, \gamma_0),\ \tilde{\alpha}_1 \sim \text{Ga}(M\gamma_0, 1)$$

with $\boldsymbol{b}^{(i)} \sim \text{IBP}(\alpha)$ and $\odot$ representing a pointwise/Hadamard product. If we consider $\gamma_0 = \hat{\alpha}_0/M$, and the limit $M \rightarrow \infty$, with $\boldsymbol{b}^{(i)}$ all ones, this corresponds to the HDP, with $\hat{\alpha}_1 \sim \text{Ga}(\hat{\alpha}_0, 1)$. Therefore, the proposed model is intimately related to the HDP, with three differences: (i) p_i is not restricted to be 1/2, which adds flexibility when modeling counts; (ii) rather than drawing $\bar{\boldsymbol{\phi}}$ and the normalization constant $\tilde{\alpha}_1$ separately, as in the HDP, in the proposed model $\boldsymbol{\phi}$ is drawn directly via $\phi_m \sim \text{Ga}(\gamma_0, 1)$, with an explicit link to the count of observations $M_i \sim \text{NegBin}(\sum_{m=1}^{M} b_m^{(i)}\phi_m, p_i)$; and (iii) the binary vectors $\boldsymbol{b}^{(i)}$ "focus" the model on a sparse subset of the mixture components, while in general, within the HDP, all mixture components have non-zero probability of occurrence for all tasks i. As demonstrated in Section IV, this focusing nature of the proposed model is important in the context of electrophysiological data.

III. Computations

The posterior distribution of model parameters is approximated via Gibbs sampling. Most of the update equations for the model are relatively standard due to conjugacy of consecutive distributions in the hierarchical model; these "standard" updates are not repeated here (see [6]). Perhaps the most important update equation is for ϕ_m, as we found this to be a critical component of the success of our inference. To perform such sampling we utilize the following lemma.

Lemma III.1. *Denote $s(n, j)$ as the Sterling numbers of the first kind [13] and $F(n, j) = (-1)^{n+j}s(n, j)/n!$ as their normalized and unsigned representations, with $F(0, 0) = 1$, $F(n, 0) = 0$ if $n > 0$, $F(n, j) = 0$ if $j > n$ and $F(n + 1, j) = \frac{n}{n+1}F(n, j) + \frac{1}{n+1}F(n, j - 1)$ if $1 \leq j \leq n$. Assuming $n \sim \text{NegBin}(\phi, p)$ is a negative binomial distributed random variable, and it is augmented into a compound Poisson representation [2] as*

$$n = \sum_{l=1}^{\ell} u_l,\ u_l \sim \text{Log}(p),\ \ell \sim \text{Pois}(-\phi \ln(1 - p)) \quad (14)$$

where $\text{Log}(p)$ is the logarithmic distribution [2] with probability generating function $G(z) = \ln(1 - pz)/\ln(1 - p)$, $|z| < p^{-1}$, then we have

$$\Pr(\ell = j|n, \phi) = R_\phi(n, j) = F(n, j)\phi^j \Big/ \sum_{j'=1}^{n} F(n, j')\phi^{j'} \quad (15)$$

for $j = 0, 1, \cdots, n$.

The proof is provided in the Appendix.

Concerning sampling ϕ_m, since $\phi_m \propto \prod_{i:b_m^{(i)}=1} \text{NegBin}(n_{im}^*; \phi_m, p_i)\text{Ga}(\phi_m; \gamma_0, 1)$, using Lemma III.1, we can first sample a latent count variable ℓ_{im} for each n_{im}^* as

$$\Pr(\ell_{im} = l|n_{im}^*, \phi_m) = R_{\phi_m}(n_{im}^*, l),\ \ l = 0, \cdots, n_{im}^*. \quad (16)$$

Since $\ell_{im} \sim \text{Pois}(-\phi_m \ln(1 - p_i))$, using the conjugacy between the gamma and Poisson distributions, we have

$$\phi_m|\{\ell_{im}, b_m^{(i)}, p_i\} \sim$$
$$\text{Ga}\left(\gamma_0 + \textstyle\sum_{i:b_m^{(i)}=1} \ell_{im}, \frac{1}{1 - \sum_{i:b_m^{(i)}=1} \ln(1-p_i)}\right). \quad (17)$$

Notice that marginalizing out ϕ_m in $\ell_{im} \sim \text{Pois}(-\phi_m \ln(1 - p_i))$ results in $\ell_{im} \sim \text{NegBin}(\gamma_0, \frac{-\ln(1-p_i)}{1-\ln(1-p_i)})$, therefore, we can use the same data augmentation technique by sampling a latent count $\tilde{\ell}_{im}$ for each ℓ_{im} and then sampling γ_0 using the gamma Poisson conjugacy as

$$\Pr(\tilde{\ell}_{im} = l|\ell_{im}, \gamma_0) = R_{\gamma_0}(\ell_{im}, l),\ \ l = 0, \cdots, \ell_{im} \quad (18)$$
$$\gamma_0|\{\tilde{\ell}_{im}, b_m^{(i)}, p_i\} \sim$$
$$\text{Ga}\left(c_0 + \textstyle\sum_{i:b_m^{(i)}=1} \tilde{\ell}_{im}, \frac{1}{d_0 - \sum_{i:b_m^{(i)}=1} \ln\left(1 - \frac{-\ln(1-p_i)}{1-\ln(1-p_i)}\right)}\right).$$

Another important parameter is $b_m^{(i)}$. Since $b_m^{(i)}$ can only be zero if $n_{im}^* = 0$ and when $n_{im}^* = 0$, $\Pr(b_m^{(i)} = 1|-) \propto \text{NegBin}(0; \phi_m, p_i)\pi_m$ and $\Pr(b_m^{(i)} = 0|-) \propto (1 - \pi_m)$, we have

$$b_m^{(i)}|\pi_m, n_{im}^*, \phi_m, p_i \sim$$
$$\text{Bernoulli}\left(\delta(n_{im}^* = 0)\frac{\pi_m(1-p_i)^{\phi_m}}{\pi_m(1-p_i)^{\phi_m} + (1-\pi_m)} + \delta(n_{im}^* > 0)\right).$$

A large p_i thus indicates a large variance-to-mean ratio on n_{im}^* and M_i. Note that when $b_m^{(i)} = 0$, the observed zero count $n_{im}^* = 0$ is no longer explained by $n_{im}^* \sim \text{NegBin}(r_m, p_i)$, this satisfies the intuition that the underlying beta-Bernoulli process is governing whether a cluster would be used or not, and once it is activated, it is r_m and p_i that control how much it would be used.

IV. Results

For these experiments we used a truncation level of $K = 40$ dictionary elements, and the number of mixture components was truncated to $M = 20$ (these truncation levels are upper bounds, and within the analysis a subset of the possible dictionary elements and mixture components are utilized). In dictionary learning, the gamma priors for $\{\eta_t\}$ and α_0 were set as $\text{Ga}(10^{-6}, 10^{-6})$. In the context of the hierarchical count and mixture modeling, we set $a_0 = b_0 = 1$, $c_0 = 0.1$ and $d_0 = 0.1$. Prior $\text{Ga}(10^{-6}, 10^{-6})$ was placed on parameter α related to the IBP. None of these parameters have been tuned, and many related settings yield similar results. In all examples we ran 6,000 Gibbs samples, with the first 3,000 discarded as burn-in (however, typically high-quality results are inferred with far fewer samples, offering the potential for computational acceleration).

A. Real data with partial ground truth

We first consider publicly available dataset[c0] hc-1. These data consist of both extracellular recordings and an intracellular recording from a nearby neuron in the hippocampus of an anesthetized rat [12]. Intracellular recordings give clean

[c0] available from http://crcns.org/data-sets/hc/hc-1

signals on a spike train from a specific neuron, providing accurate spike times for that neuron. Thus, if we detect a spike in a nearby extracellular recording within a close time period (<.5ms) to an intracellular spike, we assume that the spike detected in the extracellular recording corresponds to the known neuron's spikes.

For the accuracy analysis, we determine one cluster that corresponds to the known neuron. We consider a spike to be correctly sorted if it is a known spike and is in the known cluster or if it is an unknown spike in the unknown cluster.

We considered the widely used data d533101 and the same preprocessing from [5]. These data consist of 4-channel extracellular recordings and 1-channel intracellular recording. We used 2491 detected spikes and 786 of those spikes came from the known neuron. Accuracy rate of cluster results based on multiple methods are shown in Figure 1.The DP-DL and HDP-DL results correspond to dictionary learning applied separately to each channel (from [6]), and the Matrix DP (MDP) and FMM with the top 2 principle components without dictionary learning correspond to mixture models with the spikes observed simultaneously across all 4 channels, and the proposed model corresponds to joint dictionary learning all 4 channels, we compare DP-DL and FMM based mixture modeling (here both models employ the proposed form of dictionary learning, with the differences manifested in how the mixture component of the model is performed) . These data are relatively simple, with two clear mixture components and with the spikes observed simultaneously across all 4 channels; the Matrix DP-DL (MDP-DL) based and focused mixture model (FMM) form of the proposed model therefore yield similar results, with the gain in these results relative to DP-DL and HDP-DL deemed manifested as a result of joint dictionary learning across all channels.

Note that in Figure 1, in the context of PCA features, we considered the two principal components (similar results were obtained with the three principal components); when we considered the 20 principal components, for comparison, the results deteriorated, presumably because the higher-order components correspond to noise. An advantage of the proposed approach is that we model the noise explicitly, via the residual $\mathbf{E}_{ij}$ in (1); with PCA the signal and noise are not distinguished.

B. Handling missing data

The quantity of data acquired by a neural recording system is enormous, and therefore in many systems one first performs spike detection ([c1]for example, based on a threshold), and then a signal is extracted about each detection (a temporal window is placed around the peak of a given detection). This step is often imperfect, and significant portions of many of the spikes may be missing due to the windowed signal extraction (and the missing data are not retainable, as the original data are discarded). Conventional feature-extraction methods typically cannot be applied to such temporally clipped signals.

Returning to (1), this implies that some columns of the data $\mathbf{X}_{ij}$ may have missing entries. Conditioned on $\mathbf{D}$, $\mathbf{\Lambda}$, $\mathbf{S}_{ij}$, and $(\eta_1, \ldots, \eta_T)$, we have $\mathbf{X}_{ij} \sim \mathcal{N}(\mathbf{D}\mathbf{\Lambda}\mathbf{S}_{ij}, \text{diag}(\eta_1^{-1}, \ldots, \eta_T^{-1})$.

[c1] ~~e.g.~~

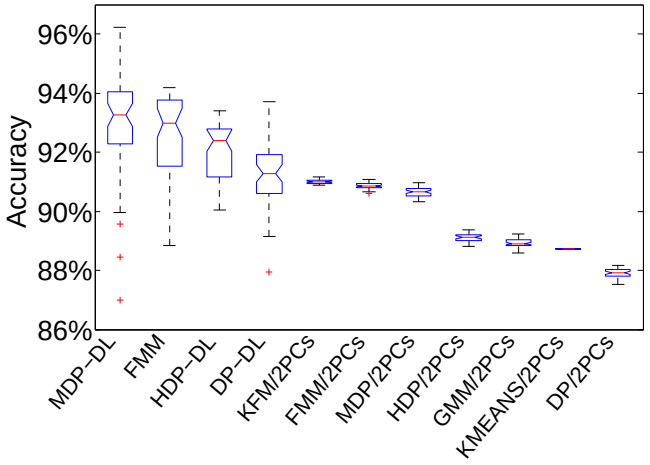


Fig. 1. *Results from testing on d533101 data [12]. KFM represents Kalman Filter Mixture method [5].GMM is Gaussian Mixture method [4]. 2 PCs denotes using the top 2 principle components, and results were indistinguishable from using the top 3 principle components. For the proposed model, dictionary learning was done as in Sec. II-A, and "FMM" corresponds to the focused mixture-model of Sec. II-C*

The missing entries of $\mathbf{X}_{ij}$ may be treated as random variables, and they are integrated out analytically within the Gaussian likelihood function. Therefore, for the case of missing data in $\mathbf{X}_{ij}$, we simply evaluate (1) at the points of $\mathbf{X}_{ij}$ for which data are observed. The columns of the dictionary $\mathbf{D}$ of course have support over the entire signal, and therefore given the inferred $\mathbf{S}_{ij}$ (in the presence of missing data), one may impute the missing components of $\mathbf{X}_{ij}$ via $\mathbf{D}\mathbf{\Lambda}\mathbf{S}_{ij}$. As long as, across all $\mathbf{X}_{ij}$, the same part of the signal is not clipped away (lost) for all observed spikes, by jointly processing all of the data (all spikes) we may infer $\mathbf{D}$, and hence infer missing data.

In practice we are less interested in observing the imputed missing parts of $\mathbf{X}_{ij}$ than we are in simply clustering the data, in the presence of missing data. By evaluating $\mathbf{X}_{ij} \sim \mathcal{N}(\mathbf{D}\mathbf{\Lambda}\mathbf{S}_{ij}, \text{diag}(\eta_1^{-1}, \ldots, \eta_T^{-1})$ only at points for which data are observed, and via the mixture model in (4), we directly infer the desired clustering, in the presence of missing data (even if we are not explicitly interested in subsequently examining the imputed values of the missing data).

To examine the ability of the model to perform clustering in the presence of missing data, we reconsider the publicly available data from Section IV-A. For the first 10% of the spike signals (300 spike waveforms), we impose that a fraction of the beginning and end of the spike is absent. The original signals are of length $T = 40$ samples. As a demonstration, for the "clipped" signals, the first 10 and the last 16 samples of the signals are missing. A clipped waveform example is shown in Figure **??**; we compare the mean estimation of the signal, and the error bars reflect one standard deviation from the full posterior on the signal. In the context of the analysis, we processed all of the data as before, but now with these "damaged"/clipped signals. We observed that 94.11% of the non-damaged signals were clustered properly (for the one neuron for which we had truth), and 92.33% of the damaged signals were sorted properly. The recovered signal

in Figure **??**(a) is typical, and is meant to give a sense of the accuracy of the recovered missing signal. The ability of the model to perform spike sorting in the presence of substantial missing data is a key attribute of the dictionary-learning-based framework.

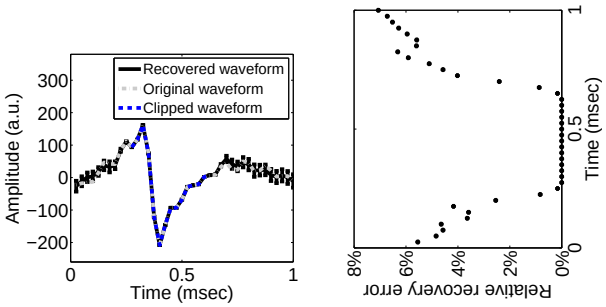


Fig. 2. *(a) Example of a clipped waveform (blue), original waveform (gray) and recovery waveform (black); the error bars reflect one standard deviation, from the posterior distribution on the underlying signal. (b) Relative recovery errors (with respect to the mean estimated signal).*

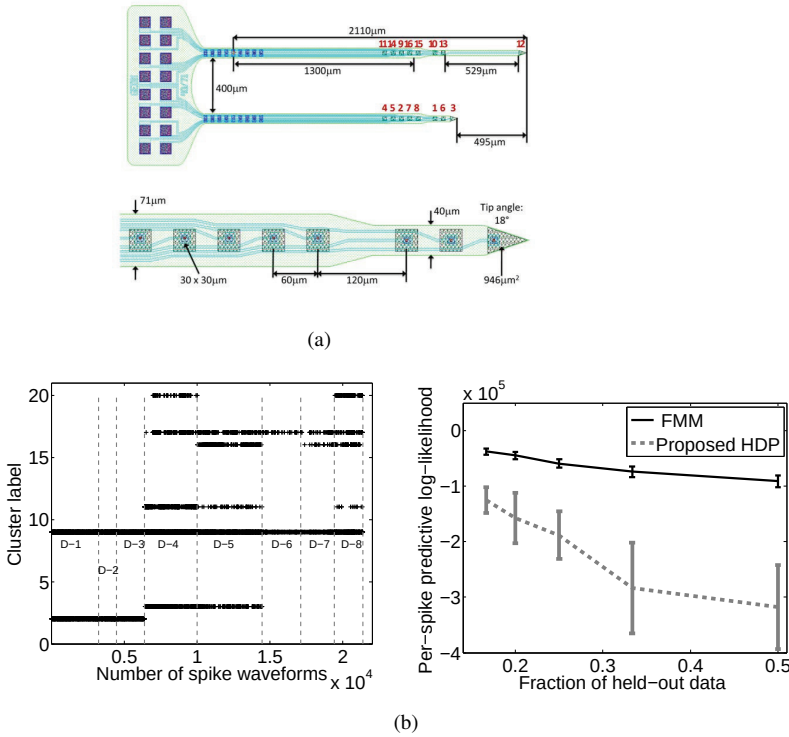


Fig. 3. *(a) Schematic of the neural recording array that was placed in the rat motor cortex. The red numbers identify the sensors, and a zoom-in of the bottom-eight sensors is shown. The sensors are ordered by the order of the read-out pads, at left. The presented data are for sensors numbered 1 to 8, corresponding to the zoomed-in region. (b) From the maximum-likelihood collection sample, the apportionment of data among mixture components (clusters). Results are shown for 45 sec recording periods, on each of 8 days. For example, D-4 reflects data on day 4. Note that while the truncation level is such that there are 20 candidate clusters (vertical axis in (b)), only an inferred subset of clusters are actually used. (c) Predictive likelihood of held-out data. The horizontal axis represents the fraction of data held out during training.*

C. Forensic analysis of new longitudinal electrophysiological data

The next dataset is new, based upon experiments we have performed with freely moving rats (institutional review board approvals were obtained). These data will be made available to the research community. NeuroNexus$^{\text{TM}}$ sensors (Figure 3(a)) were humanely placed in the motor cortex, and electrophysiological data were measured during one-hour periods on eight consecutive days, starting on the day after implant (data were collected for additional days, but the signal quality degraded after 8 days, as discussed below). Note that nearby sensors are close enough to record the signal of a single or small group of neurons, termed a single-unit event. However, all eight sensors in a line are too far separated to simultaneously record a single-unit event on all eight.

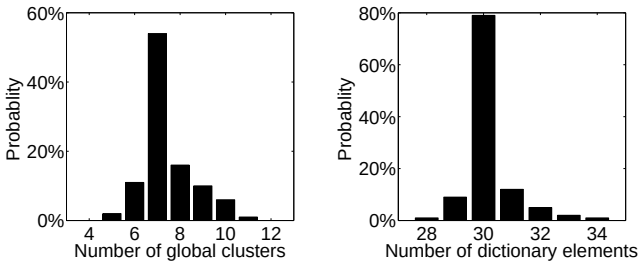
The data were bandpass filtered (0.3-3 kHz), and then all signals 3.5 times the standard deviation of the background signal were deemed detections. The peak of the detection was placed in the center of a 1.3 msec window, which corresponds to $T = 40$ samples at the recording rate. The signal $\mathbf{X}_{ij} \in \mathbb{R}^{T\times N}$ corresponds to the data measured simultaneously across all N channels within this window. Here $N = 8$, with a concentration on the data measured from the 8 channels of the zoomed-in Figure 3(a).

In Figure **??** are shown assignments of data to each of the possible clusters, for data measured across the 8 days, as computed by the proposed model ([c1]for example, for the first three days, two clusters were inferred). Results are shown for the maximum-likelihood collection sample. As a comparison to the proposed focused mixture model (FMM) of Section II-C, we also considered the simplified HDP construction discussed in Section II-D, with the $\boldsymbol{b}^{(i)}$ set to all ones (in both cases we employ the same form of dictionary learning, as in Section II-A). From Figure 3(b), it is observed that on held-out data the FMM yields improved results relative to the traditional HDP.

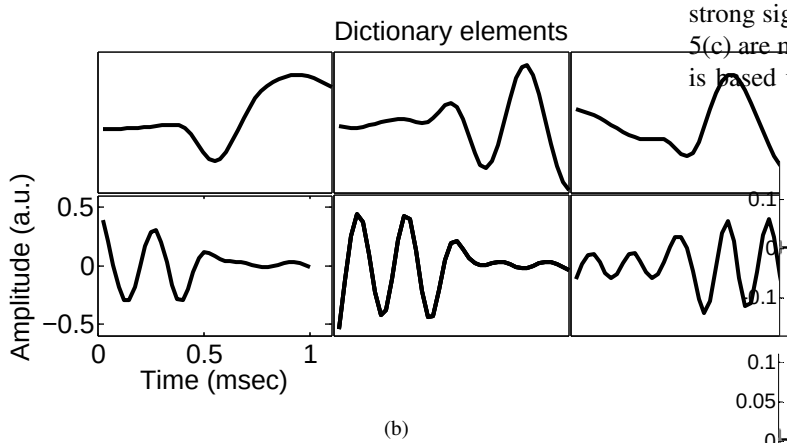
In fact, the proposed model was developed specifically to address the problem of multi-day forensic analysis of electrophysiological data, as a consequence of observed limitations of HDP (which are only partially illuminated by Figure 3(b)). Specifically, while the focused nature of the FMM allows learning of specialized clusters that occur over limited days, the "non-focused" HDP tends to merge similar but distinct clusters. This yields HDP results that are characterized by fewer total clusters, and by cluster characteristics that are less revealing of detailed neural processes. Patterns of observed neural activity may shift over a period of days due to many reasons, including cell death, tissue encapsulation, or device movement; this shift necessitates the FMM's ability to focus on subtle but important differences in the data properties over days. This ability to infer subtly different clusters is related to the focused topic model's ability [21] to discern distinct topics that differ in subtle ways. The study of large quantities of data

[c1] ~~*e.g.*~~

(8 days) makes the ability to distinguish subtle differences in clusters more challenging (the DP-DL-based model works well when observing data from one recording session, like in Figure 1, but the analysis of multiple days of data is challenging for HDP).



(a)



(b)

Fig. 4. *(a) Approximate posterior distribution on the number of global clusters (mixture components). (b) Approximate posterior distribution on the number of required dictionary elements. (c) Example inferred dictionary elements, the unit of amplitude is unitless.*

Note from Figure **??** that the number of detected signals is different for different recording days, despite the fact that the recording period reflective of these data (45 secs) is the same for all days. This highlights the need to allow modeling of different signal rates, as in our model but not emphasized in these results.

Among the parameters inferred by the model are approximate posterior distributions on the number of clusters across all days, and on the required number of dictionary elements. These approximate posteriors are shown in Figures 4(a)-4(a), and in Figure 4(b) are shown example dictionary elements. Although not shown for brevity, the $\{p_i\}$ had posterior means in excess of 0.9 .

To better represent insight that is garnered from the model, in Figure 5 are depicted the inferred properties of three of the clusters, from Day 4 (D-4 in Figure **??**). Shown are the *mean* signal for the 8 channels in the respective cluster (for the 8 channels at the bottom of Figure 3(a)), and the error bars represent one standard deviation, as defined by the estimated posterior. Note that the cluster in Figure 5(a) corresponds to a localized single-unit event, presumably from a neuron (or a coordinated small group of neurons) near the sensors associated with channels 7 and 8. The cluster in Figure 5(b) similarly corresponds to a single-unit event situated near the sensors associated with channels 3 and 6. Note the proximity of sensors 7 and 8, and sensors 3 and 6, from Figure 3(a). The HDP model uncovered the cluster in Figure 5(a), but not that in Figure 5(b).

Note Figure 5(c), in which the mean signal across all 8 channels is approximately the same (HDP also found related clusters of this type). This cluster is deemed to *not* be associated with a single-unit event, as the sensors are too physically distant across the array for the signal to be observed simultaneously on all sensors from a single neuron. This class of signals is deemed associated with an artifact or some global phenomena, due to (possibly) movement of the device within the brain, and/or because of charges that build up in the device and manifest signals with animal motion. Note that in Figures 5(a)-5(b) the error bars are relatively tight with respect to the strong signals in the set of eight, while the error bars in Figure 5(c) are more pronounced (the mean curves look clean, but this is based upon averaging thousands of signals).

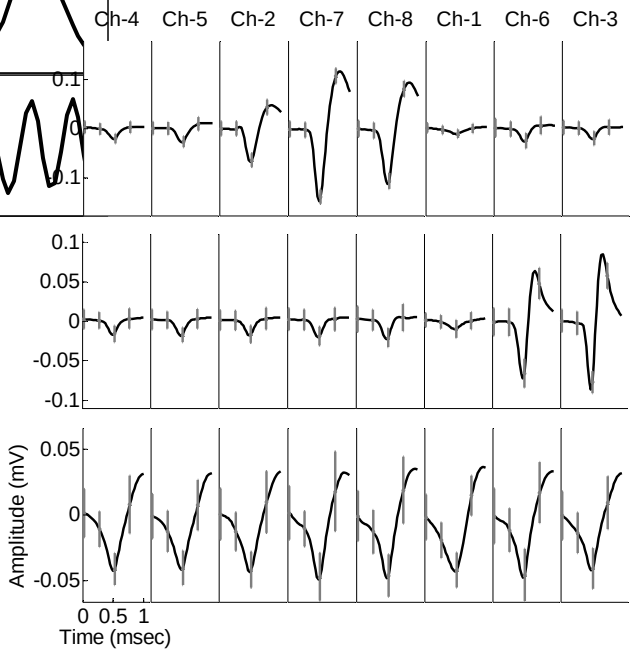


Fig. 5. *Example clusters inferred for data on the bottom 8 channels of Fig. 3(a). (a)-(b) Example of single-unit events. (c) Example of a cluster* not *attributed to a single-unit-event. The 8 signals are ordered from left to right consistent with the numbering of the 8 channels at the bottom of Figure 3(a). The blue curves represent the mean, and the error bars are one standard deviation.*

In addition to recording the electrophysiological data, video was recorded of the rat throughout. Robust PCA [22] was used to quantify the change in the video from frame-to-frame, with high change associated with large motion by the animal (this automation is required because one hour of data are collected on each day; direct human viewing is tedious and unnecessary). On Day 4, the model infers that in periods of high animal activity, 20% to 40% of the detected signals are due to single-unit events (depending on which portion of data are considered); during periods of relative rest 40% to 70% of detected signals are due to single-unit events. This suggests

that animal motion causes signal artifacts, as discussed in Section I

In these studies the total fraction of single-unit events, even when at rest, diminishes with increasing number of days from sensor implant; this may be reflective of changes in the system due to the glial immune response of the brain [3], [16]. The discerning ability of the proposed FMM to distinguish subtly different signals, and analysis of data over multiple days, has played an important role in this analysis. Further, forensic analyses like that in Figure 5 were the principal reason for modeling the data on all $N = 8$ channels jointly (the ability to distinguish single-unit events from anomalies is predicated by this multi-channel analysis).

D. Model tuning

As constituted in Section II, the model is essentially parameter free. All of the hyperparameters are set in a relatively diffuse manner (see the discussion at the beginning of Section IV), and the model infers the number of clusters and their composition with no parameter tuning required. While this may generally be viewed as a strength, there are situations for which a neuroscientist may wish to favor particular kinds of clusterings, and to have an adjustable parameter with which different solutions may be considered. All of the results presented above were manifested without any model tuning. We now discuss how one may constitute a single "knob" (parameter) that a neuroscientist may "turn" to examine different kinds of results.

In Section II-A the variance of additive noise $(e_1, \cdots, e_n)$ are controlled by the covariance $\text{diag}(\eta_1^{-1}, \cdots, \eta_T^{-1})$. If we set $\text{diag}(\eta_1^{-1}, \cdots, \eta_T^{-1}) = \omega_0^{-1}\mathbf{I}_T$, then parameter ω_0 may be tuned to control the variability (diversity) of spikes. The cluster diversity encouraged by setting different values of ω_0 in turn manifests different numbers of clusters, which a neuroscientist may adjust as desired. As an example, we consider the publicly available data from Section IV-A, and clusterings (color coded) are shown for two settings of ω_0 Figure 6. In this figure each spike is depicted in two-dimensional principal component (PC) space, taking the dominant two components; this is simply for display purposes, as here feature learning is done via dictionary learning, and in general more than two dictionary components are utilized to represent a given waveform.

The value of ω_0 defines how much of a given signal is associated with noise $\mathbf{E}_{ij}$, and how much is attributed to the term $\mathbf{D}\boldsymbol{\Lambda}\mathbf{S}_{ij}$ characterized by a summation of dictionary elements (see (1)). If ω_0 is large, then the noise contribution to the signal is small (because the noise variance is imposed to be small), and therefore the variability in the observed data is associated with variability in the underlying signal (and that variability is captured via the dictionary elements). Since the clustering is performed on the dictionary usage, if ω_0 is large we expect an increasing number of clusters, with these clusters capturing the greater diversity/variability in the underlying signal. By contrast, if ω_0 is relatively small, more of the signal is attributed to noise $\mathbf{E}_{ij}$, and the signal components modeled via the dictionary are less variable (variability is attributed to noise, not signal). Hence, as ω_0 diminishes in size we would expect fewer clusters. This phenomenon is observed in the example in Figure 6, with this representative of behavior we have observed in a large set of experiments on real data.

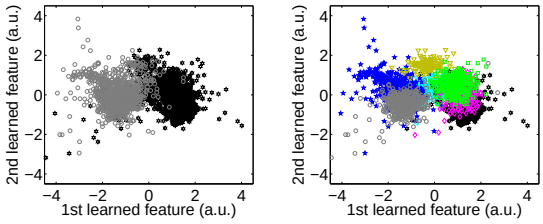
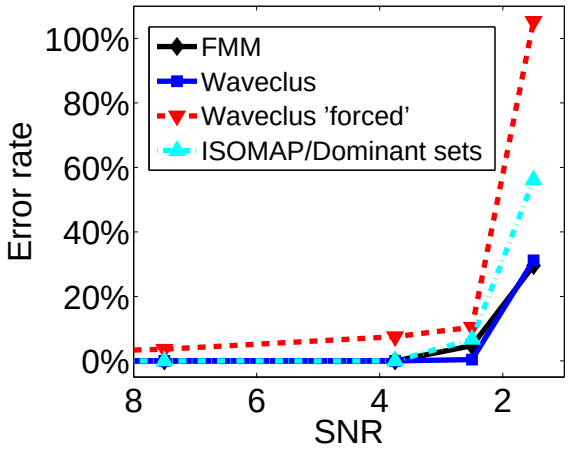
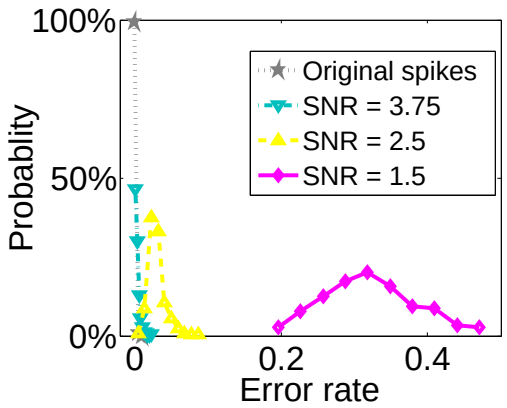


Fig. 6. *(a) Cluster results of feature spaces in the first two principle components with $\omega_0 = 10^6$, and the number of inferred clusters is two. (b) Cluster results of feature spaces in the first two principle components with $\omega_0 = 10^8$, and the number of inferred clusters is seven.*

E. Sparsely-firing problem



(a)



(b)

Fig. 8.

F. Computational requirements

The software used for the tests in this paper were written in (non-optimized) Matlab, and therefore computational efficien-

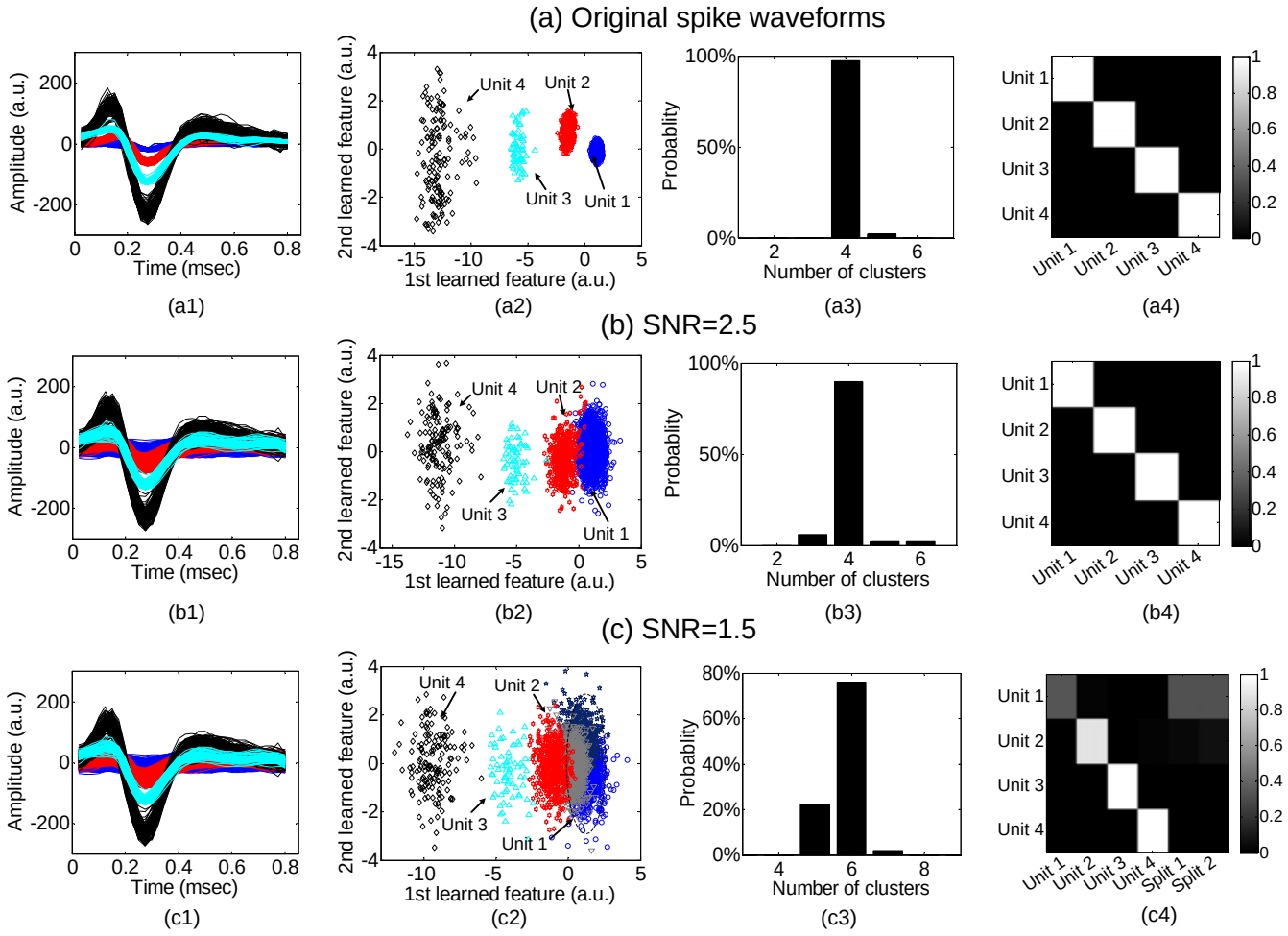


Fig. 7. stuff

cy has not been a focus. The principal motivating focus of this study concerned forensics/interpretation of spike waveforms, as discussed in Section IV-C, for which computation speed is desirable, but there is not a need for real-time processing ([c1]for example, for a prosthetic). Nevertheless, to give a sense of the computational load for the model, it takes about 20 seconds for each Gibbs sample, when considering analysis of 170800 spikes across $N = 8$ channels; computations were performed on a PC, specifically a Lenevo T420 (CPU is Inter(R) Core (TM) i7 M620 with 4 GB RAM). Significant computational acceleration may be manifested by coding in C, and via development of online methods for Bayesian inference ([c2]for example, see [20]). In the context of such online Bayesian learning one typically employs approximate variational Bayes inference rather than Gibbs sampling, which typically manifests significant acceleration [20].

V. Conclusions

A new focused mixture model (FMM) has been developed, motivated by real-world studies with longitudinal electrophysiological data, for which traditional methods like the hierarchical Dirichlet process have proven inadequate. In addition to performing "focused" clustering, the model jointly performs feature learning, via dictionary learning. The model explicitly models the count of signals within a recording period. The rate of neuron firing constitutes a primary information source [7], and therefore it is desirable that it be modeled. This rate is controlled here by a parameter p_i, and this was allowed to be unique for each recording period i. In future research one may constitute a mixture model on p_i, with each mixture component reflective of a latent neural (firing) state; one may also explicitly model the time dependence of p_i. Inference of this state could be important for decoding neural signals and controlling external devices or muscles. In future work one may also wish to explicitly account for covariates associated with animal activity [19], which may be linked to the firing rate we model here (we may regress p_i to observed covariates).

Appendix

A. Proof of Lemma 3.1

Proof: Denote $w_j = \sum_{l=1}^{j} u_l$, $j = 1, \cdots, m$. Since w_j is the summation of j iid Log(p) distributed random variables,

[c1] ~~e.g.~~
[c2] ~~e.g.~~

the probability generating function of w_j can be expressed as $G_{W_j}(z) = [\ln(1-pz)/\ln(1-p)]^j$, $|z| < p^{-1}$, thus we have

$$\begin{aligned}\Pr(w_j = m) &= G^{(m)}_{W_j}(0)/m! = \tfrac{d^m}{dz^m}[\ln(1-pz)/\ln(1-p)]^j \\ &= (-1)^m p^j j! s(m,j)/[\ln(1-p)]^j \end{aligned} \quad (19)$$

where we use the property that $[\ln(1+x)]^j = j! \sum_{n=j}^{\infty} \frac{s(n,j)x^n}{n!}$ [13]. Therefore, we have

$$\begin{aligned}\Pr(\ell = j|-) &\propto \Pr(w_j = n)\text{Pois}(j; -r\ln(1-p)) \\ &\propto (-1)^{n+j} s(n,j)/n! r^j = F(n,j) r^j. \end{aligned} \quad (20)$$

■

The values $F(n,j)$ can be iteratively calculated and each row sums to one, e.g., the 3rd to 5th rows are

$$\begin{pmatrix} 2/3! & 3/3! & 1/3! & 0 & 0 & 0 & \cdots \\ 6/4! & 11/4! & 6/4! & 1/4! & 0 & 0 & \cdots \\ 24/5! & 50/5! & 35/5! & 10/5! & 1/5! & 0 & \cdots \end{pmatrix}.$$

To ensure numerical stability when $\phi > 1$, we may also iteratively calculate the values of $R_\phi(n,j)$.

REFERENCES

[1] A. Abbott. Mind-controlled robot arms show promise. *Nature*, 2012.

[2] F. J. Anscombe. The statistical analysis of insect counts based on the negative binomial distribution. *Biometrics*, 1949.

[3] R. Biran, D.C. Martin, and P.A. Tresco. Neuronal cell loss accompanies the brain tissue response to chronically implanted silicon microelectrode arrays. *Exp. Neurol.*, 2005.

[4] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, 2006.

[5] A. Calabrese and L. Paniski. Kalman filter mixture model for spike sorting of non-stationary data. *J. Neuroscience Methods*, 2010.

[6] B. Chen, D.E. Carlson, and L. Carin. On the analysis of multi-channel neural spike data. In *NIPS*, 2011.

[7] J.P. Donoghue, A. Nurmikko, M. Black, and L.R. Hochberg. Assistive technology and robotic control using mortor cortex ensemble-based neural interface systems in humans with tetraplegia. *J. Physiol.*, 2007.

[8] A.A. Emondi, S.P. Rebrik, A.V. Kurgansky, and K.D. Miller. Tracking neurons recorded from tetrodes across time. *J. Neuro. Meth.*, 2004.

[9] J. Gasthaus, F. Wood, D. Gorur, and Y.W. Teh. Dependent Dirichlet process spike sorting. *In Advances in Neural Information Processing Systems*, 2009.

[10] D. Gorur, C. Rasmussen, A. Tolias, F. Sinz, and N. Logothetis. Modelling spikes with mixtures of factor analysers. *Pattern Recognition*, 2004.

[11] T. L. Griffiths and Z. Ghahramani. Infinite latent feature models and the Indian buffet process. In *NIPS*, 2005.

[12] D. A. Henze, Z. Borhegyi, J. Csicsvari, A. Mamiya, K. D. Harris, and G. Buzsaki. Intracellular feautures predicted by extracellular recordings in the hippocampus in vivo. *J. Neurophysiology*, 2010.

[13] N.L. Johnson, A.W. Kemp, and S. Kotz. *Univariate Discrete Distributions*. John Wiley & Sons, 2005.

[14] J. C. Letelier and P. P. Weber. Spike sorting based on discrete wavelet transform coefficients. *J. Neuroscience Methods*, 2000.

[15] M. S. Lewicki. A review of methods for spike sorting: the detection and classification of neural action potentials. *Network: Computation in Neural Systems*, 1998.

[16] D.H. Szarowski, M.D. Andersen, S. Retterer, A.J. Spence, M. Isaacson, H.G. Craighead, J.N. Turner, and W. Shain. Brain responses to micromachined silicon devices. *Brain Res.*, 2003.

[17] Y. W. Teh. Dirichlet processes. In *Encyclopedia of Machine Learning*. Springer, 2010.

[18] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical dirichlet processes. *J. Am. Stat. Ass.*, 2006.

[19] V. Ventura. Automatic spike sorting using tuning information. *Neural Computation*, 2009.

[20] C. Wang, J. Paisley, and D. Blei. Online variational inference for the hierarchical Dirichlet process. *Artificial Intelligence and Statistics*, 2011.

[21] S. Williamson, C. Wang, K.A. Heller, and D.M. Blei. The IBP compound Dirichlet process and its application to focused topic modeling. In *ICML*, 2010.

[22] J. Wright, Y. Peng, Y. Ma, A. Ganesh, and S. Rao. Robust principal component analysis: Exact recovery of corrupted low-rank matrices by convex optimization. In *Neural Information Processing Systems (NIPS)*, 2009.

[23] M. Zhou, H. Chen, J. Paisley, L. Ren, L. Li, Z. Xing, D. Dunson, G. Sapiro, and L. Carin. Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images. *IEEE Trans. Image Processing*, 2012.

[24] M. Zhou, L.A. Hannah, D.B. Dunson, and L. Carin. Beta-negative binomial process and Poisson factor analysis. In *AISTATS*, 2012.