Multichannel Electrophysiological Spike Sorting via Joint Dictionary Learning & Mixture Modeling

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

***Abstract*—We propose a methodology for joint feature learning and clustering of multichannel extracellular electrophysiological data, across multiple recording periods for action potential detection and discrimination ("spike sorting"). Our methodology improves over the previous state of the art principally in four ways. First, via sharing information across channels, we can better distinguish between single-unit spikes and artifacts. Second, our proposed "focused mixture model" (FMM) deals with units appearing, disappearing, or *reappearing* over multiple recording days, an important consideration for any chronic experiment. Third, by jointly learning features and clusters, we improve performance over previous attempts that proceeded via a two-stage learning process. Fourth, by directly modeling spike rate, we improve detection of sparsely spiking neurons. Moreover, our Bayesian methodology seamlessly handles missing data. We present state-of-the-art performance without requiring manually tuning hyperparameters, considering both a public dataset with partial ground truth and a new experimental dataset.**

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

I. Introduction

SPIKE sorting of extracellular electrophysiological data is an important problem in contemporary neuroscience, with applications ranging from brain-machine interfaces [22] to neural coding [24] and beyond. Despite a rich history of work in this area [11], [34], room for improvement remains for automatic methods. In particular, we are interested in sorting spikes from multichannel longitudinal data, where longitudinal data potentially consists of many experiments conducted in the same animal over weeks or months. Given such data, we desire spike sorting that satisfies the following:

1) achieves state-of-the-art performance,
2) copes with neurons dropping in or out over longitudinal data,
3) improves with more data,
4) is fully automatic, obviating the need for the user to manually tune many "hyperparameters", especially the number of single-units,
5) benefits from multiple electrodes,
6) is robust to artifactual noise, for example, due to movement,
7) handles "missing data", for example, due to overlapping spikes,
8) facilitates intuitive "knobs" so that an expert to fine tune performance,
9) detects sparsely firing neurons, and
10) provides an estimate of certainty.

Here we propose a Bayesian generative model and associated inference procedure; the first, to our knowledge, that satisfies all of the above desiderata to our satisfaction. Perhaps the most important advance in our present work over previous art is our joint feature learning and clustering strategy. More specifically, standard pipelines for processing extracellular electrophysiology data consist of the following steps: (i) filter the raw sensor readings, (ii) perform thresholding to "detect" the spikes, (iii) map each detected spike to a feature vector, and then (iv) cluster the feature vectors [21]. Our primary conceptual contribution to spike sorting methodologies is a novel unification of steps (iii) and (iv) that utilizes all available data in such a way as to satisfy all of the the above criteria. This *joint* dictionary learning and clustering approach improves results even for a single channel and a single recording experiment ($i.e.$, not longitudinal data). Additional localized recording channels improve the performance of our methodology by incorporating more information. More recordings allow us to track dynamics of firing over time.

Although a comprehensive survey of previous spike sorting methods is beyond the scope of this manuscript, below we provide a summary of previous work as relevant to the above listed goals.

Perhaps those methods that are most similar to ours include a number of recent Bayesian methods for spike sorting [9], [14]. One can think of our method as a direct extension of theirs with a number of enhancements. Most importantly, we learn features for clustering, rather than simply using principal components. We also incorporate multiple electrodes, assume a more appropriate prior over the number of clusters, and address longitudinal data.

Other popular methods utilize principal components analysis (PCA) [21] or wavelets [20] to find low-dimensional representations of waveforms for subsequent clustering. These methods typically require some manual tuning, for example, to choose the number of retained principal components. Moreover, these methods do not naturally handle missing data well. Finally, these methods choose low-dimensional embeddings for reconstruction and are not necessarily appropriate for

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

Manuscript received October 27, 2012.

downstream clustering.

Calabrese *et al.* [8] recently proposed a Mixture of Kalman Filters (MoK) model to explicitly deal with slow changes in waveform shape. This approach also models spike rate (and even refractory period), but it does not address our other desiderata, perhaps most importantly, utilizing multiple electrodes or longitudinal data. It would be interesting to extend that work to utilize learned time-varying dictionaries rather than principal components.

Finally, several recently proposed methods address sparsely firing neurons [2], [23]. By directly incorporating firing rate into our model and inference algorithm (see Section II-C), our approach outperforms previous methods even in the absence of manual tuning (see Section III-E).

The remainder of the manuscript is organized as follows. Section II begins with a conceptual description of our model followed by mathematical details and experimental methods for new data. Section III begins by comparing the performance of our approach to several other previous state-of-the-art methods, and then highlights the utility of a number of additional features that our method includes. Section IV summarizes and provides some potential future directions. The Appendix provides details of the relationships between our method and other related Bayesian models or methodologies.

II. Models and Analysis

A. Model Concept

Our generative model derives from knowledge of the properties of electrophysiology signals. Specifically, we assume that each waveform can be represented as a sparse superposition of several dictionary elements, or features. Rather than presupposing a particular form of those features ($e.g.$, wavelets), we *learn* features from the data. Importantly, we learn these features for the specific task at hand: spike sorting ($i.e.$, clustering). This is in contrast to other popular feature learning approaches, such as principal component analysis (PCA) or independent component analysis (ICA), which learn features to optimize a different objective function (for example, minimizing reconstruction error). Dictionary learning has been demonstrated as a powerful idea, with demonstrably good performance in a number of applications [38]. Moreover, statistical guarantees associated with such approaches are beginning to be understood [25]. Section II-B provides mathematical details for our Bayesian dictionary learning assumptions.

We *jointly* perform dictionary learning and clustering for analysis of multiple spikes. The generative model requires a prior on the number of clusters. Regardless of the number of putative spikes detected, the number of different single units one could conceivably discriminate from a single electrode is upper bounded due to the conductive properties of the tissue. Thus, it is undesirable to employ Bayesian nonparametric methods [4] that enable the number of clusters (each cluster associated with a single-unit event) to increase in an unbounded manner as the number of threshold crossings increases. We develop a new prior to address this issue, which we refer to as a "focused mixture model" (FMM). The proposed prior is also appropriate for chronic recordings, in which single units may appear for a subset of the recording days, but also disappear and reappear intermittently. Sections II-C and II-D provide mathematical details for the general mixture modeling case, and our specific focused mixture model assumptions.

We are also interested in multichannel recordings. When we have multiple channels that are within close proximity to one another, we can "borrow statistical strength" across the channels to improve clustering accuracy. Moreover, we can ascertain that certain movement or other artifacts – which would appear to be spikes if only observing a single channel – are clearly not spikes from a single neuron, as evidenced by the fact that they are observed simultaneously across all the channels, which is implausible for a single neuron. While it is possible that different neurons may fire simultaneously and be observed coincidently across multiple sensor channels, we have found that this type of observed data are more likely associated with animal motion (based on recorded video of the animal). We employ the multiple-channel analysis to distinguish single-neuron events from artifacts due to animal movement (inferred based on the electrophysiological data alone, without having to view all of the data).

Finally, we explicitly model the spike rate of each cluster. This can help address refractory issues, and perhaps more importantly, enables us to detect sparsely firing neurons with high accuracy.

Because our model is fully Bayesian, we can readily impute missing data. Moreover, by placing relatively diffuse but informed hyperpriors on our model, our approach does not require any manual tuning. And by reformulating our priors, we can derive (local) conjugacy which admits efficient Gibbs sampling. Section II-E provides details on these computations. In some settings a neuroscientist may want to tune some parameters, to tests hypotheses and impose prior knowledge about the experiment; we also show how this may be done in Section III-D.

B. 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^{th} signal observed during interval i (each j indexes a threshold crossing within a time interval i). 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 with respect to the peak energy of all the channels). In tetrode arrays [12], and related devices like those considered below, a single-unit event (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 longitudinal analysis, discussed in Section III.

To constitute data $\mathbf{X}_{ij}$, 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

centered at the spikes' energy peak average over all channels. 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 [9], we employ dictionary learning to model each $\mathbf{X}_{ij}$; however, unlike [9] 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

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

where $\mathbf{D} \in \mathbb{R}^{T\times K}$ represents a dictionary with K dictionary elements (columns), $\boldsymbol{\Lambda} \in \mathbb{R}^{K\times K}$ 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 $\boldsymbol{d}_k$, $\boldsymbol{e}_n \in \mathbb{R}^T$. We impose priors

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

where $\mathbf{I}_T$ is the $T \times T$ dimensional identity matrix 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 $\boldsymbol{\Lambda} = \text{diag}(\boldsymbol{\lambda})$, with $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_K)^T$ 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 $a_0, b_0 \in \mathbb{R}$ 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 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. We also considered using a more general noise model, with $\boldsymbol{e}_n \sim \mathcal{N}(0, \boldsymbol{\Sigma})$. These priors are typically employed in dictionary learning; see [38] for a discussion of the connection between such priors and optimization-based dictionary learning.

C. Mixture modeling

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

$$\boldsymbol{s}_{ijn} \sim \mathcal{N}(\boldsymbol{\mu}_{z_{ijn}}, \boldsymbol{\Omega}_{z_{ijn}}^{-1}), \qquad z_{ij} \sim \textstyle\sum_{m=1}^M \pi_m^{(i)}\delta_m, \tag{3}$$

$$(\boldsymbol{\mu}_{mn}, \boldsymbol{\Omega}_{mn}) \sim G_0(\mu_0, \beta_0, W_0, \nu_0) \tag{4}$$

where G_0 is a normal-Wishart distribution with μ_0 a K dimension vector of zeros, $\beta_0 = 1$, W_0 is a K dimensional identity matrix, and $\nu_0 = K$. The other parameters: $\pi_m^{(i)} > 0$, $\sum_{m=1}^M \pi_m^{(i)} = 1$, 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, 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 $\boldsymbol{\pi}^{(i)} = (\pi_1^{(i)}, \ldots, \pi_M^{(i)})^T$.

D. Focused Mixture Model

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. Let $b_m^{(i)}$ be a binary variable indicating whether interval i uses mixture component m. Let $\hat{\phi}_m^{(i)}$ correspond to the relative probability of including mixture component m in interval i, which is related to the firing rate of the single-unit corresponding to this cluster during that interval. Given this, the probability of cluster m in interval i is

$$\pi_m^{(i)} = \tfrac{1}{Z} b_m^{(i)} \hat{\phi}_m^{(i)} \tag{5}$$

where $Z = \sum_{m'=1}^M b_{m'}^{(i)} \hat{\phi}_{m'}^{(i)}$ is the normalizing constant to ensure that $\sum_m \pi_m^{(i)} = 1$. To finalize this parameterization, we further assume the following priors on $b_m^{(i)}$ and $\hat{\phi}_n^{(i)}$:

$$\hat{\phi}_m^{(i)} \sim \text{Ga}(\phi_m, p_i/(1-p_i)),$$
$$\phi_m \sim \text{Ga}(\gamma_0, 1), \ p_i \sim \text{Beta}(a_0, b_0) \tag{6}$$

$$b_m^{(i)} \sim \text{Bern}(\nu_m),$$
$$\nu_m \sim \text{Beta}(\alpha/M, 1), \ \gamma_0 \sim \text{Ga}(c_0, 1/d_0) \tag{7}$$

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 clustering across all time intervals. The reasons for the choices of these various priors is discussed in Section IV-B, 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 [16] as $M \to \infty$.

Note that we explicitly model the probability of spiking for each cluster component in each time interval. 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 greater than one hundred—but 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.

We refer to this as a focused mixture model (FMM) because the ν_m defines the probability with which cluster m is observed, and via the prior in (7) the model only "focuses" on a small number of clusters, those with large ν_m. Further, as discussed below, the parameter ϕ_m controls the firing rate of neuron/cluster m, and that is also modeled. When the $\pi_m^{(i)}$ are

modeled via a Dirichlet process (DP) [4], and the matrix of multi-channel data are modeled jointly, we refer to the model as matrix DP (MDP). If a DP is employed separately on each channel the results are simply termed DP. The hierarchical DP model in [9] for $\pi_m^{(i)}$ the model is referred to as HDP.

E. 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 [9]). 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 II.1. *Denote $s(n,j)$ as the Sterling numbers of the first kind [19] 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 [3] as*

$$n = \sum_{l=1}^{\ell} u_l,\ u_l \sim \text{Log}(p),\ \ell \sim \text{Pois}(-\phi \ln(1-p)) \tag{8}$$

where $\text{Log}(p)$ is the logarithmic distribution [3] with probability generating function $G(z) = \ln(1-pz)/\ln(1-p)$, $|z| < p^{-1}$, then we have

$$\Pr(\ell = j|n,\phi) = R_\phi(n,j) = F(n,j)\phi^j \Big/ \sum_{j'=1}^{n} F(n,j')\phi^{j'} \tag{9}$$

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

The proof is provided in the Appendix.

Let the total set of data measured during interval i be represented $\boldsymbol{\mathcal{D}}_i = \{\mathbf{X}_{ij}\}_{j=1}^{M_i}$, where M_i is the total number of events during interval i. 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 $M_i = \sum_{m=1}^{M} n_{im}^*$. To sample ϕ_m, since $\text{p}(\phi_m|\boldsymbol{p}, n_{\cdot m}^\star) \propto \prod_{i:b_m^{(i)}=1} \text{NegBin}(n_{im}^*;\phi_m,p_i)\text{Ga}(\phi_m;\gamma_0,1)$ (see Appendix IV-B for details), using Lemma II.1, we can first sample a latent count variable ℓ_{im} for each n_{im}^* as

$$\Pr(\ell_{im} = l|n_{im}^*,\phi_m) = R_{\phi_m}(n_{im}^*,l),\quad l = 0,\cdots,n_{im}^*. \tag{10}$$

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

$$\phi_m|\{\ell_{im}, b_m^{(i)}, p_i\} \sim \text{Ga}\left(\gamma_0 + \textstyle\sum_{i:b_m^{(i)}=1} \ell_{im}, \frac{1}{1-\sum_{i:b_m^{(i)}=1}\ln(1-p_i)}\right). \tag{11}$$

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

$$\Pr(\tilde{\ell}_{im} = l|\ell_{im},\gamma_0) = R_{\gamma_0}(\ell_{im},l),\quad l = 0,\cdots,\ell_{im} \tag{12}$$

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

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.

F. Data Acquisition and Pre-processing

In this work we use two datasets, the popular "hc-1" dataset[1] and a new dataset 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. Six animals were used in this study. Each animal was trained, under food restriction (15 g/animal/day, standard hard chow), on a simple lever-press-and-hold task until performance stabilized and then taken in for surgery. Each animal was implanted with four different silicon microelectrodes (NeuroNexus Technologies; Ann Arbor, MI; custom design) in the forelimb region of the primary or supplementary motor cortex. Each electrode contains up to 16 independent recording sites, with variations in device footprint and recording site position ($e.g.$, Figure 3(a)). 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). The recordings were conducted in a high walled observation chamber under freely-behaving conditions. 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, in the device in Figure 3(a), 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

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

concentration on the data measured from the 8 channels of the zoomed-in Figure 3(a).

G. Evaluation Criteria

We use several different criteria to evaluate the performance of the competing methodologies. Let F_p and F_n denote the total number of false positives and negatives for a given neuron, respectively, and let $\#_w$ denote the total number of detected waveforms. We define:

$$\text{Accuracy} = \left\{1 - \frac{F_p + F_n}{\#_w}\right\} \times 100\%. \tag{13}$$

For synthetic missing data, as in Section III-C, we compute the relative recovery error (RRE):

$$\text{RRE} = \frac{\left\|\boldsymbol{X} - \hat{\boldsymbol{X}}\right\|}{\|\boldsymbol{X}\|} \times 100\%, \tag{14}$$

where $\boldsymbol{X}$ is the true waveform, $\hat{\boldsymbol{X}}$ is the estimated waveform, and $\|\cdot\|$ indicates the L_2 or Frobenius norm depending on context. When adding noise, we compute the signal-to-noise ratio (SNR) as in [26]:

$$\text{SNR} = \frac{A}{2SD_{noise}}, \tag{15}$$

where A denotes the peak-to-peak voltage difference of the mean waveform and SD_{noise} is the standard deviation of the noise. The noise level is estimated by mean absolute deviation.

To simulate a lower SNR in the sparse spiking experiments, we took background signals from the dataset where no spiking occurred and scale them by α and add them to our detected spikes; this gives a total noise variance of $\sigma^2(1+\alpha^2)$, and we set the SNR to 2.5 and 1.5 for these experiments.

III. 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 focused mixture model, 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 Indian Buffet Process (see Appendix IV-B for details). 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 hc-1. These data consist of both extracellular recordings and an intracellular recording from a nearby neuron in the hippocampus of an anesthetized rat [17]. Intracellular recordings give clean 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 (< 0.5ms) to an intracellular spike, we assume that the spike detected in the extracellular recording corresponds to the known neuron's spikes.

We considered the widely used data d533101 and the same preprocessing from [8]. 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 of cluster results based on multiple methods are shown in Figure 1. We consider several different clustering schemes and two different strategies for learning low-dimensional representations of the data. Specifically, we learn low-dimensional representations using either: dictionary learning (DL) or the first two principal components (PCs) of the matrix consisting of the concatenated waveforms. For the multichannel data, we stack each waveform matrix to yield a vector, and concatenate stacked waveforms to obtain the data matrix upon which PCA is run. Given this representation, we consider several different clustering strategies: (i) Matrix Dirichlet Process (MDP), which implements a DP on the $\mathbf{X}_{ij}$ *matrices*, as opposed to previous DP approaches on vectors [9], [14], (ii) focused mixture model (as described above), (iii) Hierarchical DP (HDP), (iv) independent DP (both these versions of DP are from [9]), (v) Mixture of Kalman filters (MoK) [8], (vi) Gaussian mixture models (GMM) [7], and (vii) K-means (KMEANS) [21]. Although we do not consider all pairwise comparisons, we do consider many options. Note that all of the DL approaches are novel. It should be clear from Figure 1 that dictionary learning enhances performance over principal components for each clustering approach. Specifically, all DL based methods outperform all PC based methods. Moreover, sharing information across channels, as in MDP and FMM (both novel methodologies), seems to further improve performance. The ordering of the algorithms is essentially unchanged upon using a different number of mixture components or a different number of principal components.

In Figure 2, we visualize the waveforms in the first 2 principle components for comparison. In Figure 2a, we show ground truth to compare to the results we get by clustering from the K-means algorithm shown in Figure 2b and the clustering from the GMM shown in Figure 2c. We observe that both K-means and GMM work well, but due to the constrained feature space they incorrectly classify some spikes (marked by arrows). However, the proposed model, shown in Figure 2(d), which incorporates dictionary learning with spike sorting, infers an appropriate feature space (not shown) and more effectively clusters the neurons.

Note that in Figure 1 and 2, 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 explicitly

distinguished.

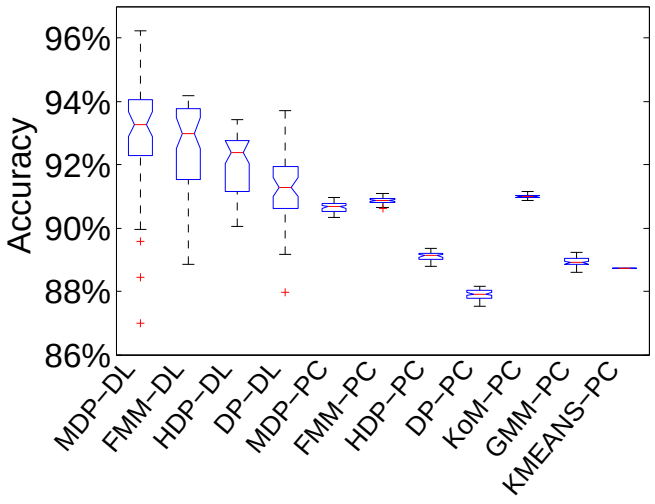


Fig. 1. Accuracy of the various methods on d533101 data [17]. All abbreviations are explained in the main text (Section III-A). Note that dictionary learning dominates performance over principal components. Moreover, modeling multiple channels (as in MDP and FMM) dominates performance over operating on each channel separately.

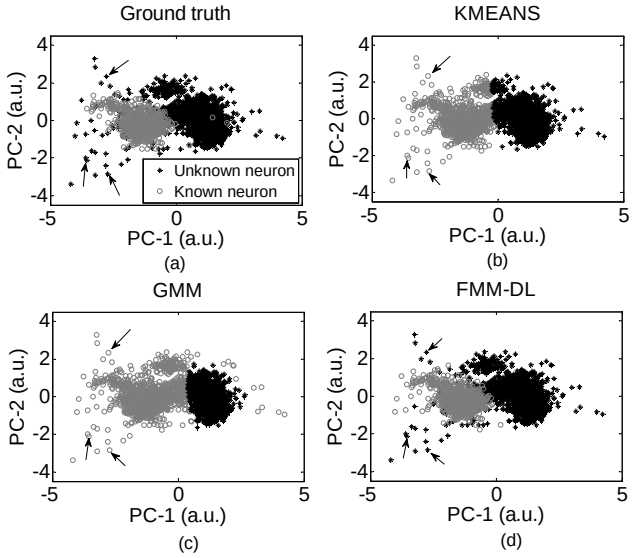
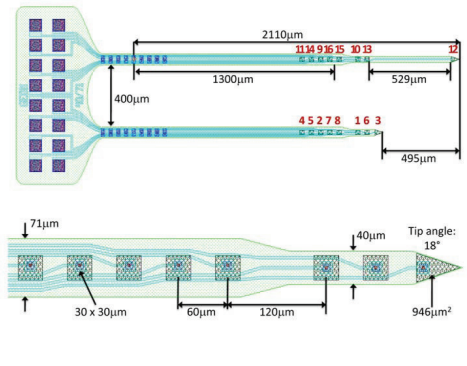


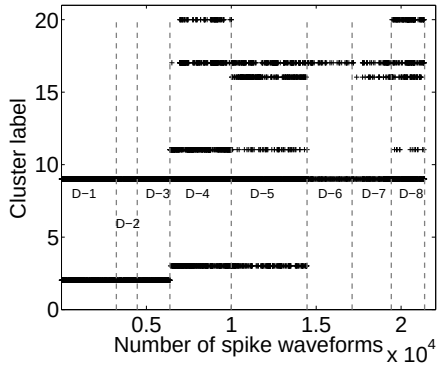
Fig. 2. Clustering results shown in the 2 PC space of the various methods on d533101 data [17]. All abbreviations are explained in the main text (Section III-A). "Known neuron" denotes waveforms associated with the neuron from the cell with the intracellular recording, and "Unknown neuron" refers to all other detected waveforms. Note that all methods are shown in the first two PCs for visualization, but that the FMM-DL shown in (d) is jointly learning the feature space and clustering.

B. Longitudinal analysis of electrophysiological data

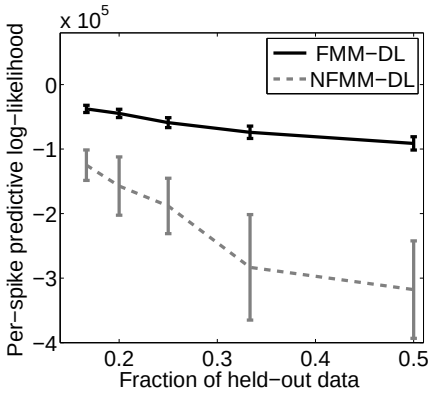
Figure 3(b)(a) shows the recording probe used for the analysis of the rat motor cortex data. Figure 3(b) shows assignments of data to each of the possible clusters, for data measured across the 8 days, as computed by the proposed model (for example, for the first three days, two clusters were inferred). Results are shown for the maximum-likelihood collection sample. As a comparison to FMM-DL, we also considered



(a)



(b)



(c)

Fig. 3. Longitudinal data analysis of the rat motor cortex data. (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 on any given day. (c) Predictive likelihood of held-out data. The horizontal axis represents the fraction of data held out during training. FMM-DL dominates NFMM-DL on these data.

the non-focused mixture model (NFMM-DL) methodology discussed in Section IV-B, with the $\boldsymbol{b}^{(i)}$ set to all ones (in both cases we employ the same form of dictionary learning, as in Section II-B). From Figure 3(c), it is observed that on held-out data the FMM-DL yields improved results relative to the NFMM-DL.

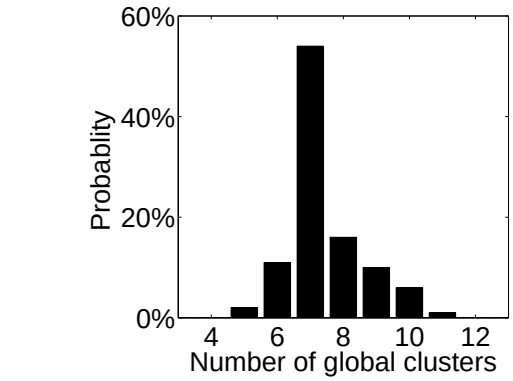
In fact, the proposed model was developed specifically to address the problem of multi-day longitudinal analysis of electrophysiological data, as a consequence of observed limitations of HDP (which are only partially illuminated by Figure 3(c)). Specifically, while the focused nature of the FMM-DL allows learning of specialized clusters that occur over limited days, the "non-focused" HDP-DL 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-DL'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 [35] to discern distinct topics that differ in subtle ways. The study of large quantities of data (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).

Note from Figure 3(b) 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 firing rates, as in our model but not emphasized in these results.

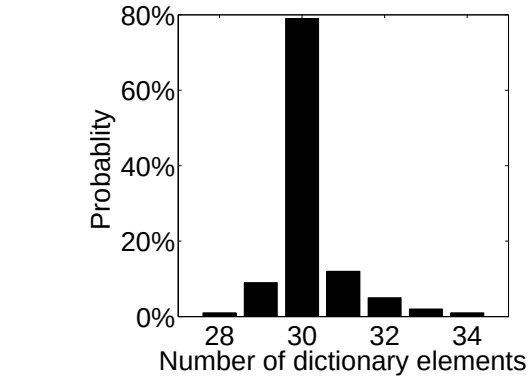
Among the parameters inferred by the model are approximate posterior distributions on the number of clusters across all days, and on the required number of dictionary elements. These approximate posteriors are shown in Figures 4(a) and 4(b), and Figure 4(c) shows example dictionary elements. Although 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, Figure 5 depicts the inferred properties of three of the clusters, from Day 4 (D-4 in Figure 3(b)). 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 the top row of Figure 5 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 the middle row of Figure 5 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 the top row of Figure 5, but not that in the middle row of Figure 5 (not shown).

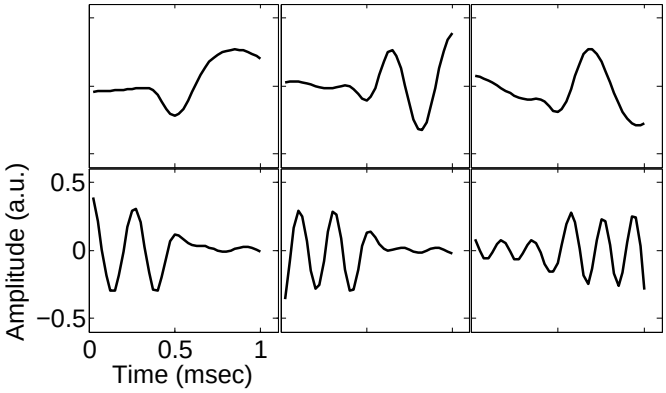
Note the bottom row of Figure 5, in which the mean signal



(a)



(b)



(c)

Fig. 4. Posteriors and dictionaries from rat motor cortex data (the same data as in Figure 3).(a) Approximate posterior distribution on the number of global clusters (mixture components). (b) Approximate posterior distribution of the number of dictionary elements. (c) Examples of inferred dictionary elements; amplitudes of dictionary elements are unit less.

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, (possibly) due to movement of the device within the brain, and/or because of charges that build up in the device and manifest signals with animal motion (by examining separate video recordings, such electrophysiological data occurred when the animal constituted significant and

abrupt movement, such as heading hitting the sides of the cage, or during grooming). Note that in the top two rows of Figure 5 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 smooth , but this is based upon averaging thousands of signals).

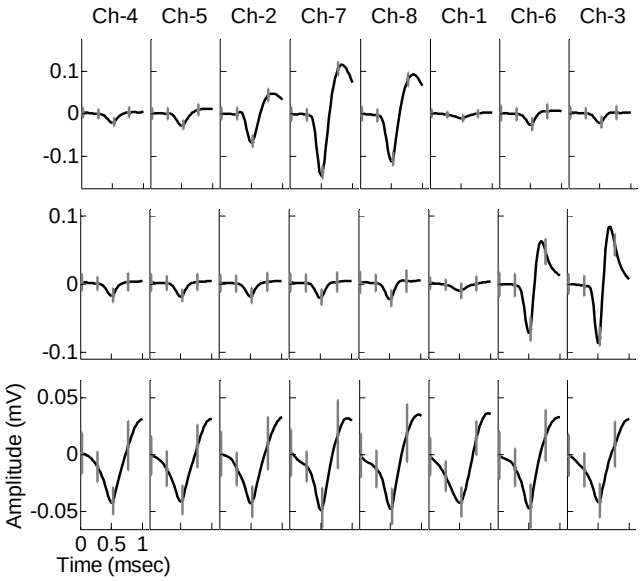


Fig. 5. Example clusters inferred for data on the bottom 8 channels of Fig. 3(a). (a)-(b) Example of single-unit events. (c) Example of a cluster *not* attributed to a single-unit-event. The 8 signals are ordered from left to right consistent with the numbering of the 8 channels at the bottom of Figure 3(a). The black 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 the experiment. Robust PCA [36] 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 useful 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 [6], [27]. The discerning ability of the proposed FMM-DL to distinguish subtly different signals, and analysis of data over multiple days, has played an important role in this analysis. Further, longitudinal 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 on this multi-channel analysis).

C. 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 (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 $\mathbf{X}_{ij} \sim \mathcal{N}(\mathbf{D}\mathbf{\Lambda}\mathbf{S}_{ij}, \text{diag}(\eta_1^{-1}, \ldots, \eta_T^{-1})$. 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 retained 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 III-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 6(a); 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 6(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.

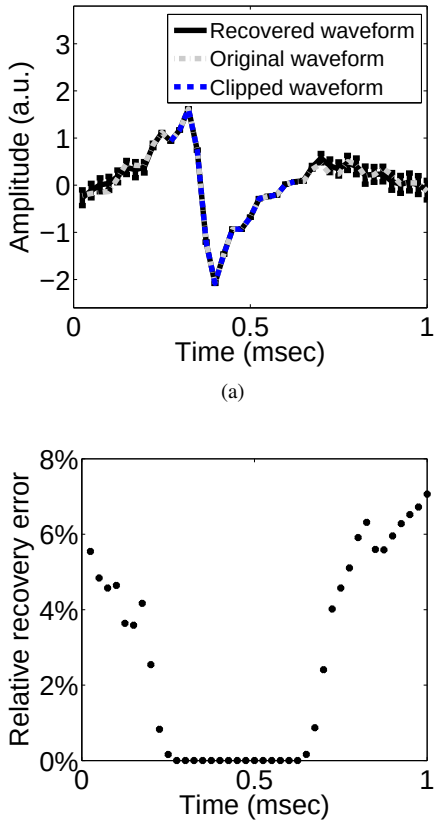


Fig. 6. Our generative model elegantly addresses missing data. (a) Example of a clipped waveform from the publicly available data (blue), original waveform (gray) and recovery waveform (black); the error bars reflect one standard deviation from the posterior distribution on the underlying signal. (b) Relative errors (with respect to the mean estimated signal). Note that we only show part of the waveform for visualization purposes.

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 III), and the model infers the number of clusters and their composition with no parameter tuning required. Thus, our code runs "out-of-the-box" to yield state-of-the-art accuracy on the dataset that we tested. And yet, an expert experimentalist could desire different clustering results, further improving the performance. Because our inference methodology is based on a biophysical model, all of the hyperparameters have natural and intuitive interpretations. Therefore, adjusting the performance is relatively intuitive. Although 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-B 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 III-A, and clusterings (color coded) are shown for two settings of ω_0 in Figure 7. In this figure, each spike is depicted in two-dimensional learned feature space, taking two arbitrary features (because features are not inherently ordered); 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 $\mathbf{D\Lambda S}_{ij}$ 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 7, with this representative of behavior we have observed in a large set of experiments on the rat motor cortex data.

E. Sparsely Firing Neurons

Recently, several manuscripts have directly addressed spike sorting in the present of sparsely firing neurons [2], [23]. We operationally define a sparsely firing neuron as a neuron whose spike count has significantly fewer spikes than the other isolated neurons. Based on reviewer recommendations, we assessed the performance of FMM-DL in such regimes utilizing the following synthetic data. First, we extracted spike waveforms from four clusters from the new dataset discussed in Section II-F. We excluded all waveforms that did not clearly separate (Figure 8(a1)) to obtain clear clustering criteria (Figure 8(a2)). There were 2592, 148, 506, and 64 spikes in the first, second, third, and fourth cluster, respectively. Then, we added real noise—as described in section II-G—to each waveform at two different levels to obtain increasingly noisy and less-well separated clusters (Figure 8(b1), (b2), (c1), and (c2)). We applied FMM-DL, Wave-clus [23] and Wave-clus "forced" (in which we hand tune the parameters to obtain optimal results) and ISOMAP dominant sets [2] to all three signal-to-noise ratio (SNR) regimes to assess our relative performance with the following results.

The third column of Figure 8 shows the posterior estimate of the number of clusters for each of the three scenarios. As long as SNR is relatively good, for example, higher than 2 in this simulation, the posterior number of clusters inferred by FMM-DL correctly has its maximum at four clusters. Similarly, for the good and moderate SNR regimes, the confusion matrix is essentially a diagonal matrix, indicating that FMM-DL assigns spikes to the correct cluster. Only in the poor SNR regime

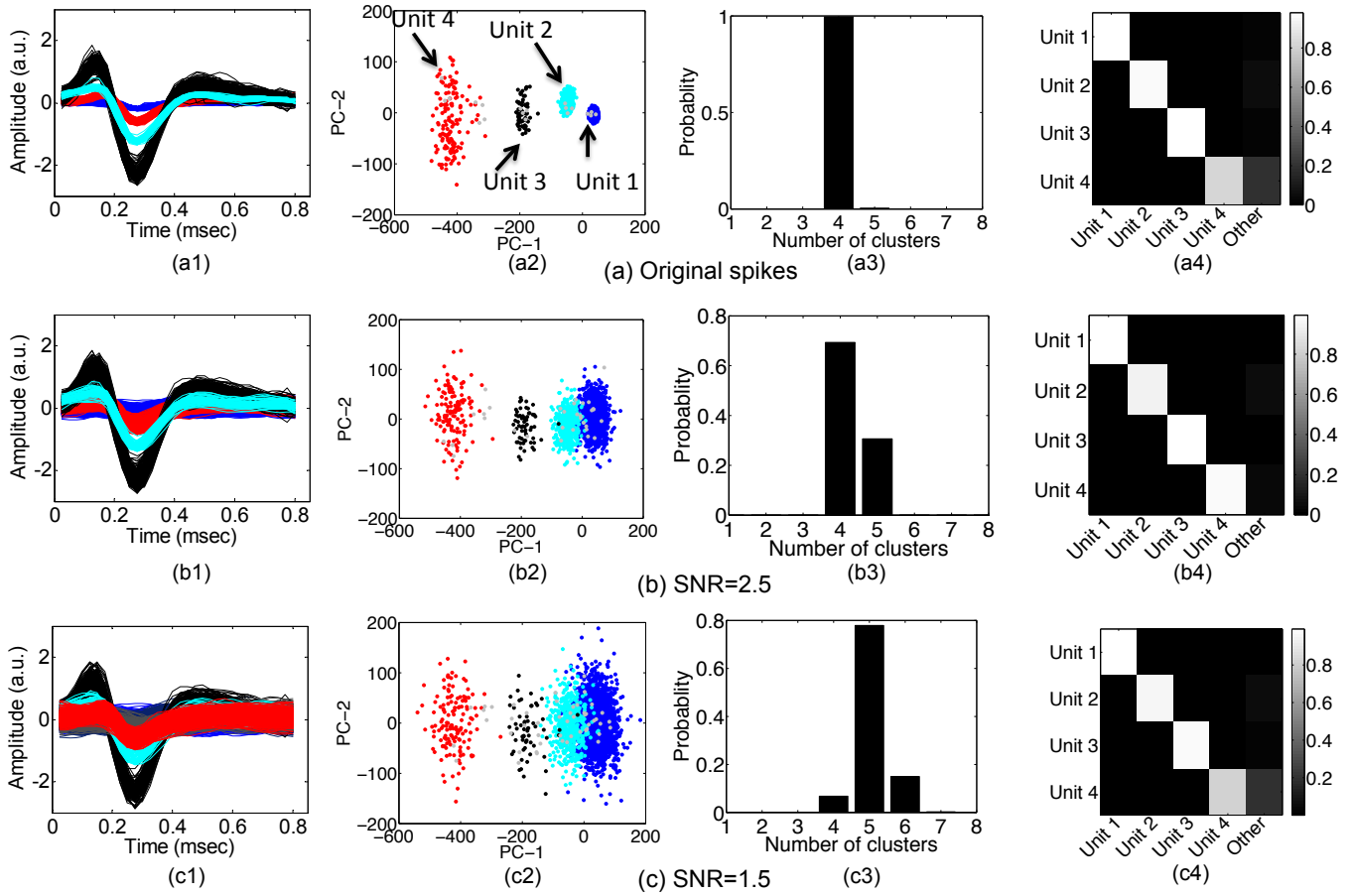


Fig. 8. Sparse firing results on synthetic data based on the Pittsburgh dataset. The three rows correspond to three different signal-to noise ratio (SNR) levels: (a) 1, (b) 1.5, and (c) 2.5. The four columns correspond to: (1) cluster results of spike waveforms with colors representing different clusters, (2) plots of learned features based on cluster result, (3) approximate posterior distribution of cluster numbers, and (4) confusion matrix heatmap. Note that we accurately recover all the sparsely spiking neurons except the sparsest one in the noisiest regime.

(SNR=1.5), does the posterior move away from the truth. This occurs because Unit 1 becomes over segmented, as depicted in (c2). (c4) shows that only this unit struggles with assignment issues, suggestive of the possibility of a post-hoc correction if desired.

Figure 9(a) compares the performance of FMM-DL to previously proposed methods. Even after fine-tuning the Wave-clus method to obtain its optimal performance on these data, FMM-DL yields a better accuracy. In addition to obtaining better point-estimates of spiking, via our Bayesian generative model, we also obtain posteriors over all random variables of our model, including number of spikes per unit. Figure 9(b) and (c) show such posteriors, which may be used by the experimentalist to assess data quality.

F. Computational requirements

The software used for the tests in this paper were written in (non-optimized) Matlab, and therefore computational efficiency has not been a focus. The principal motivating focus of this study concerned interpretation of longitudinal spike waveforms, as discussed in Section III-B, for which computation speed is desirable, but there is not a need for real-time processing (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 Intel(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 (for example, see [32]). In the context of such online Bayesian learning one typically employs approximate variational Bayes inference rather than Gibbs sampling, which typically manifests significant acceleration [32].

IV. Discussion

A. Summary

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, which significantly improves performance over principal components. We explicitly model the count of signals within a recording period by

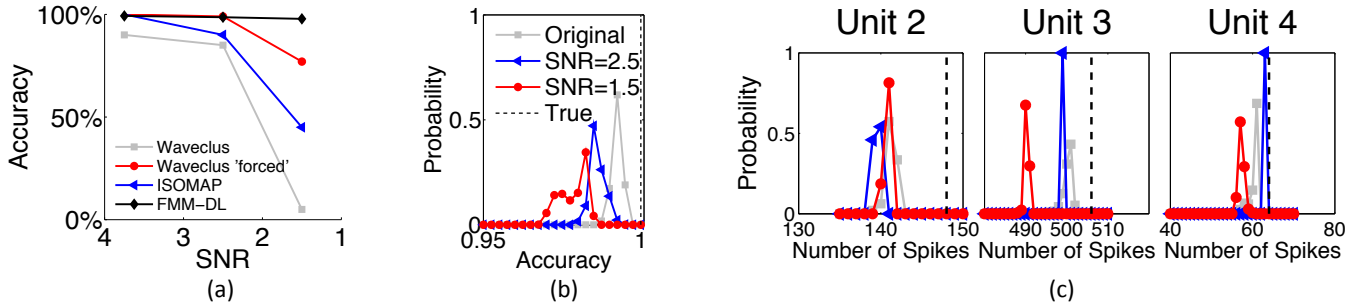


Fig. 9. Performance analysis in the sparsely firing neuron case on synthetic data based on the Pittsburgh dataset. (a) Accuracy comparisons based on the cluster results under the various SNR. (b) Approximate posterior distributions of error rate for FMM-DL in the different SNR levels. (c) Approximate posterior distributions of spike waveform number for the unit 2, unit 3. and unit 4 under the various SNR regimes.

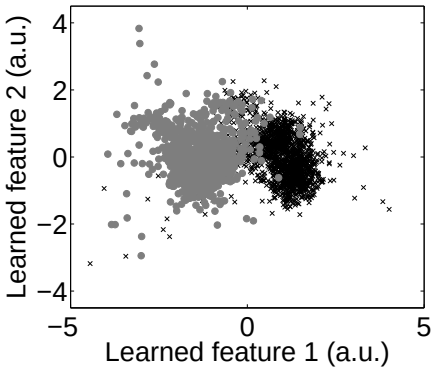


Fig. 7. Effect of manually tuning ω_0 to obtain a different number of features for the rat motor cortex data. (a) Waveforms projected down onto two learned features based on cluster result with $\omega_0 = 10^6$, the number of inferred clusters is two. (b) Same as (a) with $\omega_0 = 10^8$; the number of inferred clusters is seven.

p_i. The rate of neuron firing constitutes a primary information source [10], and therefore it is desirable that it be modeled. This rate is controlled here by a parameter $\phi_m^{(i)}$, and this was allowed to be unique for each recording period i.

B. Future Directions

In future research one may constitute a mixture model on $\phi_m^{(i)}$, with each mixture component reflective of a latent neural (firing) state; one may also explicitly model the time dependence of $\phi_m^{(i)}$, as in the Mixture of Kalmans work [8]. 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 [31], which may be linked to the firing rate we model here (we may regress p_i to observed covariates).

In the context of modeling and analyzing electrophysiological data, recent work on clustering models has accounted for refractory-time violations [8], [9], [14], 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 [9], [14] 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 III, a key component of the proposed method is that it allows us to distinguish single-unit (spike) events from other phenomena.

Perhaps the most important feature of spike sorting methods that we have not explicitly included in this model is "overlapping spikes" [1], [5], [13], [18], [30], [33], [37]. Preliminary analysis of our model in this regime (not shown), inspired by reviewer comments, demonstrated to us that while the FMM-DL as written is insufficient to address this issue, a minor modification to FMM-DL will enable "demixing" overlapping spikes. We are currently pursuing this avenue. Neuronal bursting—which can change the waveform shape of a neuron—is yet another possible avenue for future work.

ACKNOWLEDGEMENT

The research reported here was supported by the Defense Advanced Research Projects Agency (DARPA) under the HIST program, managed by Dr. Jack Judy. The findings and opinions in this paper are those of the authors alone.

APPENDIX

A. Connection to Bayesian Nonparametric Models

The use of nonparametric Bayesian methods like the Dirichlet process (DP) [9], [14] 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 $\mathcal{O}(\alpha \log S)$ [28], 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) [29] shares all mixture components, which may undermine inference of subtly different clusters.

In this paper we integrate dictionary learning and clustering for analysis of electrophysiological data, as in [9], [15]. However, as an alternative to utilizing a method like DP or HDP [9], [14] 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 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) [16] with the Dirichlet distribution, similar to [35], 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 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 (for example, spikes) measured within a given recording interval. To our knowledge, this is the first time the firing rate of electrophysiological data is modeled jointly with clustering *and* 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 FMM-DL method, considering new measured (experimental) electrophysiological data.

B. Relationship to Dirichlet priors

A typical prior for $\boldsymbol{\pi}^{(i)}$ is a symmetric Dirichlet distribution [15],

$$\boldsymbol{\pi}^{(i)} \sim \text{Dir}(\tilde{\alpha}_0/M, \ldots, \tilde{\alpha}_0/M). \tag{16}$$

In the limit, $M \rightarrow \infty$, this reduces to a draw from a Dirichlet process [9], [14], 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 from $\text{DP}(\tilde{\alpha}_0 G_0)$, we may consider the hierarchical Dirichlet process (HDP) [29] as

$$\boldsymbol{\pi}^{(i)} \sim \text{DP}(\tilde{\alpha}_1 G) \ , \quad G \sim \text{DP}(\tilde{\alpha}_0 G_0) \tag{17}$$

The HDP methodology imposes that the $\{\boldsymbol{\pi}^{(i)}\}$ share the same set of "atoms" $\{\boldsymbol{\mu}_{mn}, \boldsymbol{\Omega}_{mn}\}$, 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 [9].

These models have limitations in that the inferred number of clusters grows with observed data (here the clusters are ideally connected to neurons, the number of which will not necessarily grow with 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 (for example, spike rate) are modeled explicitly.

C. Other Formulations of the FMM

Let the total set of data measured during interval i be represented $\boldsymbol{\mathcal{D}}_i = \{\mathbf{X}_{ij}\}_{j=1}^{M_i}$, where M_i is the total number of events during interval i. 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 the priors for $b_m^{(i)}$ and $\hat{\phi}_m^{(i)}$ given in Eqs. (6) and (7). 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 methodology 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 [39] 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

$$(n_{i1}^*, \ldots, n_{iM}^*) \sim \text{Mult}(M_i; \pi_1^{(i)}, \ldots, \pi_M^{(i)}) \tag{18}$$

$$\pi_m^{(i)} = b_m^{(i)} \hat{\phi}_m^{(i)} / \textstyle\sum_{m'=1}^{M} b_{m'}^{(i)} \hat{\phi}_{m'}^{(i)} \tag{19}$$

with $\hat{\phi}_m^{(i)}$, $\{\phi_m\}$, $\{b_m^{(i)}\}$, and $\{p_i\}$ constituted as in (6)-(7). 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 (19) is now used within the mixture model in (4).

As a consequence of the manner in which $\hat{\phi}_m^{(i)}$ is drawn in (6), and the definition of $\boldsymbol{\pi}^{(i)}$ in (19), for *any* $p_i \in (0, 1)$, the proposed model imposes

$$\boldsymbol{\pi}^{(i)} \sim \text{Dir}(b_1^{(i)}\phi_1, \ldots, b_M^{(i)}\phi_M) \tag{20}$$

D. Additional Connections to Other Bayesian Models

Eq. (20) demonstrates that the proposed model is a generalization of (16). 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 [35], 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 $M_i \sim$ 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 $1/2$ as in [35], 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 [10]. 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 (6)-(7), 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

$$\boldsymbol{\pi}^{(i)} \sim \text{Dir}(\tilde{\alpha}_1(\boldsymbol{b}^{(i)} \odot \bar{\boldsymbol{\phi}})) \tag{21}$$
$$\bar{\boldsymbol{\phi}} = (\bar{\phi}_1, \cdots, \bar{\phi}_M) \sim \text{Dir}(\gamma_0, \cdots, \gamma_0),\ \tilde{\alpha}_1 \sim \text{Ga}(M\gamma_0, 1)$$

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 $\hat{\alpha}_1 \sim \text{Ga}(\hat{\alpha}_0, 1)$. We call such a model the non-focused mixture model (NFMM). 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 III, this focusing nature of the proposed model is important in the context of electrophysiological data.

E. 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, 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

$$\begin{aligned}\Pr(w_j = m) &= G_{W_j}^{(m)}(0)/m! = \tfrac{d^m}{dz^m}[\ln(1-pz)/\ln(1-p)]^j \\ &= (-1)^m p^j j! s(m, j)/[\ln(1-p)]^j \end{aligned} \tag{22}$$

where we use the property that $[\ln(1+x)]^j = j!\sum_{n=j}^{\infty}\frac{s(n,j)x^n}{n!}$ [19]. Therefore, we have

$$\begin{aligned}\Pr(\ell = j|-) &\propto \Pr(w_j = n)\text{Pois}(j; -r\ln(1-p)) \\ &\propto (-1)^{n+j}s(n, j)/n!r^j = F(n, j)r^j. \end{aligned} \tag{23}$$

■

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] D. A. Adamos, N. A. Laskaris, E. K. Kosmidis, and G. Theophilidis. NASS: an empirical approach to spike sorting with overlap resolution based on a hybrid noise-assisted methodology. *Journal of neuroscience methods*, 190(1):129–42, 2010.

[2] D. A. Adamos, N. A. Laskaris, E. K. Kosmidis, and G. Theophilidis. In quest of the missing neuron: spike sorting based on dominant-sets clustering. *Computer methods and programs in biomedicine*, 107(1):28–35, 2012.

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

[4] C. Antoniak. Mixtures of dirichlet processes with applications to bayesian nonparametric problems. *The Annals of Statistics*, 2:1152–1174, 1974.

[5] I. Bar-Gad, Y. Ritov, E. Vaadia, and H. Bergman. Failure in identification of overlapping spikes from multiple neuron activity causes artificial correlations. *Journal of Neuroscience Methods*, 107(1-2):1–13, 2001.

[6] 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.

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

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

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

[10] 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.

[11] G. T. Einevoll, F. Franke, E. Hagen, C. Pouzat, and K. D. Harris. Towards reliable spike-train recordings from thousands of neurons with multielectrodes. *Current opinion in neurobiology*, 22(1):11–7, 2012.

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

[13] F. Franke, M. Natora, C. Boucsein, M. H. J. Munk, and K. Obermayer. An online spike detection and spike classification algorithm capable of instantaneous resolution of overlapping spikes. *Journal of computational neuroscience*, 29(1-2):127–48, 2010.

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

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

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

[17] D. A. Henze, Z. Borhegyi, J. Csicsvari, A. Mamiya, K. D. Harris, and G. Buzsaki. Intracellular features predicted by extracellular recordings in the hippocampus in vivo. *J. Neurophysiology*, 2000.

[18] J. A. Herbst, S. Gammeter, D. Ferrero, and R. H. R. Hahnloser. Spike sorting with hidden Markov models. *Journal of neuroscience methods*, 174(1):126–34, 2008.

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

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

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

[22] M. A. L. Nicolelis and M. A. Lebedev. Principles of neural ensemble physiology underlying the operation of brain-machine interfaces. *Nature Reviews Neuroscience*, 10(7):530–540, 2009.

[23] C. Pedreira, Z. J. Martinez, M. J. Ison, and R. Quian Quiroga. How many neurons can we see with current spike sorting algorithms? *Journal of neuroscience methods*, 211(1):58–65, 2012.

[24] D. Rieke, F .and Warland, R. De Ruyter Van Steveninck, and W. Bialek. *Spikes: Exploring the Neural Code*, volume 20. MIT Press, 1997.

[25] D. A. Spielman, H. Wang, and J. Wright. Exact Recovery of Sparsely-Used Dictionaries. *Computing Research Repository*, 2012.

[26] S. Suner, M. R. Fellows, C. Vargas-Irwin, G. K. Nakata, and J. P. Donoghue. Reliability of signals from a chronically implanted, silicon-based electrode array in non-human primate primary motor cortex. *IEEE Trans. on Neural Systems and Rehabilitation Engineering*, 13(4):524–541, 2005.

[27] 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 micro-machined silicon devices. *Brain Res.*, 2003.

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

[29] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, pages 101:1566–1581, 2006.

[30] C. Vargas-Irwin and J. P. Donoghue. Automated spike sorting using density grid contour clustering and subtractive waveform decomposition. *Journal of neuroscience methods*, 164(1):1–18, 2007.

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

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

[33] X. Wang and A. McCallum. Topics over time: a non-markov continuous-time model of topical trends. *The Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 424–433, 2006.

[34] B. C. Wheeler. Multi-unit neural spike sorting. *Annals of Biomedical Engineering*, 19(5), 1991.

[35] 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.

[36] 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 *NIPS*, 2009.

[37] J. Zhang, Z. Ghahramani, and Y. Yang. A probabilistic model for online document clustering with application to novelty detection. *Proceedings of Neural Information Processing Systems*, 2004.

[38] M. Zhou, H. Chen, J. Paisley, L. Ren, L. Li, Z. Xing, David B. Dunson, G. Sapiro, and L. Carin. Nonparametric Bayesian Dictionary Learning for Analysis of Noisy and Incomplete Images. *IEEE Trans. Image Processing*, 21(1):130–144, 2012.

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

REBUTTAL

General comments

We would like to thank the reviewers for their very helpful comments additional insights. Below, we have appended the reviewer comments, along with our responses in red. Quotes from the revised text are in *red italics*.

A. Response to Reviewer 2

1) Major Concerns:

a. The authors have strengthened their manuscript by comparing the performance of their method with 2 other algorithms. However the incorporation of a simulated white Gaussian (independent and identically distributed) noise profile in the comparison does not adequately support the manuscript.
Thank you for this suggestion. We have revised our analysis of the sparsely firing data according to your suggestion. In particular, rather than white noise, we incorporate real electrophysiology noise. Please see the revised §III-E for details.

2) Minor Concerns:

a. A definition of the SNR representation (obviously linear) that the authors follow, is absent.
We have added the definition in Section II-G.

b. (Abstract) We propose a "construction" . . . Our "construction" The use of term "algorithm", "method" seems more plausible.
replaced "construction" with "methodology" throughout.

c. "(Page 1, L30).. more recordings simply improve our performance." Replace our with its, or revise.
Replaced with "*Additional localized recording channels improve the performance of our methodology by incorporating more information. More recordings allow us to track dynamics of firing over time.*"

B. Response to reviewer 3

1) General Comments: I will refer to a sentence in the manuscript, e.g., on page 1 in the right column on line 40 as p1 rc 40. Given the huge scientific effort that was invested into the spike sorting problem and the fact that the principal problems are still not solved, the abstract and introduction of this manuscript are very bold. The authors claim to have solved the following problems/their method has the [below] ... attributes. However, in my view, most of these statements are not backed up by the authors with sufficient and convincing results showing that they actually solved the problems. These claims are addressed individually below. The authors should either weaken their claims or add significantly more results.

Thank you for pointing out our inappropriately bold language. We have revised the language and/or performed additional experiments to address each of your specific claims. Note in particular how we have revised §I. We have appended below part of it for ease:

In particular, we are interested in sorting spike from multichannel longitudinal data, where longitudinal data might consist of many experiments conducted in the same animal over weeks or months. Given such data, we desire a spike sorting that satisfies the following desiderata:

1) *achieves state-of-the-art performance,*
2) *copes with neurons dropping in or out over longitudinal data,*
3) *improves with more data,*
4) *is fully automatic, obviating the need for the user to manually tune many "hyperparameters", especially the number of single-units,*
5) *benefits from multiple electrodes,*
6) *is robust to artifactual noise, due to movement, for example,*
7) *elegantly handles "missing data", for example, due to overlapping spikes,*
8) *facilitates intuitive "knobs" so that an expert to fine tune performance,*
9) *detects sparsely firing neurons, and*
10) *provides an estimate of certainty.*

Here we propose a Bayesian generative model and associated inference procedure; the first, to our knowledge, that satisfies all of the above desiderata to our satisfaction.

2) Major Comments:

a. **Claim a)** Their method is fully automatic with absolutely no human interactions necessary.
While our method does not *require* manual fine tuning to achieve state-of-the-art performance, it can still *benefit* from manual tuning of the hyperparemeters. We have clarified in the text that a benefit of our generative model is "knobs" that are intuitive, such that if an expert disagrees with the algorithm's performance, it can easily be adapted appropriately. In particular, in addition to the revision to §I appended above, we also modified §III-D as shown below:
Thus, our code runs "out-of-the-box" to yield state-of-the-art accuracy on the dataset that we tested. And yet, expert experimentalist could desire different clustering results, further improving the performance. Because our inference methodology is based on a biophysical model, all of the hyperparameters have natural and intuitive interpretations. Therefore, adjusting the performance is relatively intuitive. Although 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.

b. **Claim b)** The methods performance improves with number of recording channels.
We regrettably failed to justify this claim sufficiently in the main text. In addition to softening the claim in §I (we just require that the method benefits from multiple electrodes) we add the below text to §II-A which describes how we use multiple channels to assist in detecting events that are not isolated single unit spikes.
Moreover, we can ascertain that certain movement or other artifacts—which would appear to be spikes if only observing a single channel—are clearly not spikes from a single neuron, as evidenced by the fact that they are observed across all the channels, which is implausible for a single neuron. Note that such a spike looking event across all channels could, for instance, be a synchronized spike across many neighboring neurons, or movement. While without video or some other evidence of movement it is difficult to distinguish between these two situations, neither setting provides much evidence for a spike from the isolated unit that we believe to be recording from. For recording in awake behaving animals, such artifacts can be quite common.

c. **Claim c)** The methods performance improves with length of recordings.
We have softened this claim. Note that Figure 3(c) shows that as we increase the amount of data, the per-spike predictive log-likelihood increases.

d. **Claim d)** The method is robust to movement artifacts by distinguishing them from single unit spikes by "sharing information across channels".
Thank you for pointing this out. The above appended new quote from §II-A hopefully clarifies this point.

e. **Claim e)** The method handles missing data.
We agree that cutting far into the waveform can improve spiking performance. We have modified the requirement to be "*elegantly handles missing data*". In a sense, we are leveraging the insight that you point out. Essentially, our approach is robust to the waveform being truncated. Rather than requiring that the investigator explicitly truncate the waveforms, she can leave them untruncated, and the algorithm can use those waveforms that are truncated, along with those that are not.

f. **Claim f)** The method can deal with non-stationary data over days and even weeks.

g. **Claim h)** The method can find neurons that appear in the data.

h. **Claim i)** The method can re-find neurons that disappeared from the data.
We have softened these claims. We now request that our method "*copes with neurons dropping in or out over longitudinal data*".

Coping is a relatively mild requirement that our methodology explicitly addresses.

i. **Claim g)** The method is good in finding sparsely firing neurons. ... A definition of "sparsely firing" is missing. The number of spikes in the individual clusters in fig. 8 are not given!
We have revised the simulations to include real noise, rather than white noise. While the numerical results differ, the qualitative results are nearly identical. We have also now defined "sparsely firing" neurons in §III-E:
We operationally define a sparsely firing neuron as a neuron whose spike count has significantly fewer spikes than the other isolated neurons.

j. I do not agree with the way the authors compare their performance to that of PCA. Clustering in PC space is particularly effected by alignment errors and the number of PCs chosen. Spike alignment should be carried out with upsampling the waveforms and also "easy" procedures to automatically chose the number of PCs exist. This could greatly improve the reference methods performance (and proper spike alignment might even improve the authors methods performance). Also, how were the parameters for the reference methods chosen, e.g., how was the "k" for k-means determined?
We agree that the PCA analysis that other methods utilize could be substantially improved. Our intent in comparison was to compare with state of the art algorithms that are most commonly employed. Our numerical results were essentially identical upon keeping a larger number of principle components. We have appended the following sentence: We have added the below to §III-A:
The ordering of the algorithms is essentially unchanged upon using a different number of mixture components or a different number of principal components.
What was the space in which the PCs were computed? Were the individual channels concatenated for that? Two and also 3 PCs might be far too few for a 4 channel recording with several neurons present.
We have now clarified in the §III-A:
Specifically, we learn low-dimensional representations using either: dictionary learning (DL) or the first two principal components (PCs) of the matrix consisting of the concatenated waveforms. For the multichannel data, we stack each waveform matrix to yield a vector, and concatenate stacked waveforms to obtain the data matrix upon which PCA is run.
To my knowledge the hc-1 d533101 data set used by the authors has a sampling rate of 10kHz (at least this is written in the file d533101.xml downloaded from the cited cncrs website). If they cut out 40 samples per channel, that results in a 4ms long piece of data. But the waveform in Fig.3 is only 1ms long?
We truncated the waveform for visualization purposes. We have clarified in the caption of Figure 6:
Note that we only show part of the waveform for visualization purposes.

k. The authors claim that the joint dictionary learning outperforms classical feature selection techniques like PCA and wavelets but they do not show this part of their results.
We failed to clarify that DL outperforms PC is indicated in Figure 1. We add a clarifying sentence to the text:
Specifically, all DL based methods outperform all PC based methods.

3) Minor Comments:

a. In the introduction the authors give a list of features an ideal spike sorter would have. This list coincides with the features the authors claim for their method. However, as pointed out by the authors, e.g., in the discussion, an IDEAL spike sorter would have to have even more features: resolving overlapping spike, dealing with bursts, FAST (run time), estimate of the sorting performance! The last point the authors can actually claim for their method, however, they do not.
Thank you! We have added: "*provides an estimate of certainty.* " to the list in §I.

b. "I) A. Privious Art", 2nd , "...of theirs with a number enhancements." (p1 rc 40) is missing an "of".
fixed

c. I have not read the term "longitudinal data" so far in the spike sorting literature. The authors might want to explain that term in the introduction.
We have now added to §I, "*In particular, we are interested in sorting spike from multichannel longitudinal data, where longitudinal data potentially consists of many experiments conducted in the same animal over weeks or months.*"

d. Missing data is usually not a problem in spike sorting and it is not directly clear what part would be missing. The authors might want to explain that term in the introduction.
We have clarified: "*elegantly handles "missing data", for example, due to overlapping spikes*" in the list in §I.

e. In fig.8 and 9 the term "Pittsburgh dataset" is used. I guess this is the data set the authors created in this paper. They should name it differently.
We lack the creativity to think of a more informative and concise name.

f. The authors use the hc-1 data set because it is "widely used". If that is so, they should provide more than one citation using it and compare their performance to that of other publications.
Several of the methods compared in §III-A have previously been utilized on these data.

g. p6 lc 44: the authors report an accuracy of 94.11% for the "undamaged" waveforms. How does this relate to fig. 1? Is that simply one value from the distribution shown there for MDP-DL? What happens to the PCA performance if all signals are clipped this way?
The dataset is 90% "undamaged" waveforms and 10% clipped waveforms. This is the MAP sample from the algorithm and is expected to be similar to Fig. 1 because of the majority of the data is the same, but is run on a combination of clipped and complete data. If all of the signals were clipped in this way, the PCA kmeans performance drops to 86.27%. If only thresholded waveforms have been collected, then we may have waveforms of different length and PCA cannot run naively.

h. On page 4 is two times the same footnote with a different number.
fixed

i. There is a spelling error in citation [16] "features". Also the citations year was 2000 not 2010.
fixed