Scalable Spike Source Localization in Extracellular Recordings using Amortized Variational Inference

Cole L. Hurwitz
School of Informatics
University of Edinburgh, United Kingdom
cole.hurwitz@ed.ac.uk

Kai Xu
School of Informatics
University of Edinburgh, United Kingdom
kai.xu@ed.ac.uk

Akash Srivastava
MIT–IBM Watson AI Lab
Cambridge, United States
Akash.Srivastava@ibm.com

Alessio Paolo Buccino
Department of Informatics
University of Oslo, Oslo, Norway
alessiob@ifi.uio.no

Matthias Hennig
School of Informatics
University of Edinburgh, United Kingdom
m.hennig@ed.ac.uk

Abstract

Extracellular recordings using modern, dense probes provide detailed footprints of action potentials (spikes) from thousands of neurons simultaneously. Inferring the activity of single neurons from these recordings, however, is a complex blind source separation problem, complicated both by the high intrinsic data dimensionality and large data volume. Despite these complications, dense probes can allow for the estimation of a spike's source location, a powerful feature for determining the firing neuron's position and identity in the recording. Here we present a novel, generative model for inferring the source of individual spikes given observed electrical traces. To allow for scalable, efficient inference, we implement our model as a variational autoencoder and perform amortized variational inference. We evaluate our method on biophysically realistic simulated datasets, showing that our method outperforms heuristic localization methods such as center of mass and can improve spike sorting performance significantly. We further apply our model to real data to show that it is an effective, interpretable tool for analyzing large-scale extracellular recordings.

1 Introduction

Extracellular recordings, which measure local potential changes due to ionic currents flowing through cell membranes, are an essential source of data in experimental and clinical neuroscience. The most prominent signals in these recordings originate from spikes, the all or none events neurons produce in response to inputs and transmit as outputs to other neurons.

Traditionally, a small number of electrodes are used to monitor spiking activity from a few neurons simultaneously. Recent progress in microfabrication now allows for extracellular recordings from thousands of neurons using microelectrode arrays (MEAs), which have thousands of closely spaced electrodes (channels) [8, 2, 9, 1, 28, 41, 25, 20, 7]. These recordings provide insights that cannot be obtained by pooling multiple single-electrode recordings [22]. This is a significant development as it

Preprint. Under review.

enables systematic investigations of large circuits of neurons to better understand their function and structure, as well as how they are affected by injury, disease, and pharmacological interventions [15].

On dense MEAs, each recording channel may record spikes from multiple, nearby neurons, while each neuron may leave an extracellular footprint on multiple channels. Inferring the spiking activity of individual neurons, a task called *spike sorting*, is therefore a challenging blind source separation problem, complicated by the large volume of recorded data [36]. Despite the challenges presented by spike sorting large-scale recordings, its importance cannot be overstated as it has been shown that isolating the activity of individual neurons is essential to understanding brain function [27]. Recent efforts have concentrated on providing scalable spike sorting algorithms for large scale MEAs and already several methods can be used for recordings taken from hundreds to thousands of channels [33, 24, 5, 40, 17, 21]. However, scalability, and in particular automation, of spike sorting pipelines remains challenging [36].

One strategy for spike sorting on dense MEAs is to spatially localize detected spikes before classification. In theory, spikes from the same neuron should be localized to the same region of the recording area (near the cell body of the firing neuron), providing a discriminatory, low-dimensional feature set for each spike that can be utilized with efficient density-based clustering algorithms to sort large data sets with tens of millions of detected spikes [17, 21]. These location estimates, while useful for spike sorting, can also be utilized in downstream analysis, for instance to register recorded neurons with anatomical information [17, 32]. Existing pre-sorting localization methods use simple heuristics such as a center of mass calculation. There have been attempts to estimate locations post-sorting, but these either use supervised learning and are thus unsuitable when the origin of the spikes is unknown or are too computationally expensive to apply to large-scale datasets [38, 26, 4].

In this work, we present an unsupervised, model-based approach to spike localization on dense recording devices (less than ~ 50 μm between channels). Along with our model, we provide a data augmentation scheme and an amortized variational inference method implemented with a variational autoencoder (VAE) [6, 23, 37]. After training, our method allows for localization of one million spikes in approximately 37 seconds, enabling real-time analysis of massive extracellular datasets.

We quantitatively evaluate our method on biophysically realistic simulated data demonstrating that our localization performance is significantly better than the center of mass baseline and can lead to high-accuracy spike sorting results across multiple probe geometries and noise levels. We also show that our trained VAE can generalize to recordings on which it was not trained. Finally, to demonstrate the applicability of our method to real data, we assess our method qualitatively on data samples from a Neuropixels probe and from a BioCam4096 recording platform.

2 Background

Spike localization We start with introducing relevant notation. First, we define the identities and positions of neurons and channels. Let $\boldsymbol{n} \coloneqq \{n_i\}_{i=1}^M$, be the set of M neurons in the recording and $\boldsymbol{c} \coloneqq \{c_j\}_{j=1}^N$, the set of N channels on the MEA. The position of a neuron, n_i, can be defined as $p_{n_i} \coloneqq (x_{n_i} y_{n_i}, z_{n_i}) \in \mathbb{R}^3$ and similarly the position of a channel, c_j, $p_{c_j} \coloneqq (x_{c_j}, y_{c_j}, z_{c_j}) \in \mathbb{R}^3$. We further denote $p_{\boldsymbol{c}} \coloneqq \{p_{c_j}\}_{j=1}^N$ to be the position of all N channels on the MEA. In our treatment of this problem, the neuron and channel positions are single points that represent the centers of the somas and the centers of the channels, respectively. These positions are relative to the origin, which we set to be the center of the MEA. For the neuron, n_i, let $\boldsymbol{s}_i \coloneqq \{s_{i,k}\}_{k=1}^{K_i}$, be the set of spikes detected during the recording where K_i is the total number of spikes fired by n_i. The recorded extracellular waveform of $s_{i,k}$ on a channel, c_j, can then be defined as $w_{i,k,j} \coloneqq \{r_{i,k,j}^{(0)}, r_{i,k,j}^{(1)}, ..., r_{i,k,j}^{(t)}, ..., r_{i,k,j}^{(T)}\}$ where $r_{i,k,j}^{(t)} \in \mathbb{R}$ and $t = 0, \ldots, T$. The set of waveforms recorded by each of the N channels of the MEA during the spike, $s_{i,k}$, is defined as $\boldsymbol{w}_{i,k} \coloneqq \{w_{i,k,j}\}_{j=1}^N$. Finally, for the spike, $s_{i,k}$, the point source location can be defined as $p_{s_{i,k}} \coloneqq (x_{s_{i,k}}, y_{s_{i,k}}, z_{s_{i,k}})$.

The problem we attempt to solve can now be stated as follows: *Localizing a spike, $s_{i,k}$, is the task of finding the corresponding point source location, $p_{s_{i,k}}$, given the observed waveforms $\boldsymbol{w}_{i,k}$ and the channel positions, $p_{\boldsymbol{c}}$.*

We make the assumption that the true location of the point source location, $p_{s_{i,k}}$ is actually the location of the firing neuron's soma, p_{n_i}. Given the complex morphological structure of many

neurons, this assumption may not be correct, but it provides a simple way to assess localization performance and improve future models.

Center of mass Modern spike sorting software localizes spikes on MEAs using the center of mass or barycenter method [34, 29, 17, 21]. We summarize the traditional steps for localizing a spike, $s_{i,k}$ using this method. First, let us define $\alpha_{i,k,j} \coloneqq \min_t w_{i,k,j}$ to be the negative amplitude peak of the waveform, $w_{i,k,j}$, generated by $s_{i,k}$ and recorded on channel, c_j. We consider the negative peak amplitude as a matter of convention since spikes are defined as inward currents. Then, let $\boldsymbol{\alpha}_{i,k} \coloneqq (\alpha_{i,k,j})_{j=1}^{N}$ be the vector of all amplitudes generated by $s_{i,k}$ and recorded by all N channels on the MEA.

To find the center of mass of a spike, $s_{i,k}$, the first step is to determine the central channel for the calculation. This central channel is set to be the channel which records the minimum amplitude for the spike, $c_{j_{min}} \coloneqq c_{\text{argmin}_j \alpha_{i,k,j}}$ The second and final step is to take the L closest channels to $c_{j_{min}}$ and compute, $\hat{x}_{s_{i,k}} = \dfrac{\sum_{j=1}^{L+1}(x_{c_j})|\alpha_{i,k,j}|}{\sum_{j=1}^{L+1}|\alpha_{i,k,j}|}, \hat{y}_{s_{i,k}} = \dfrac{\sum_{j=1}^{L+1}(y_{c_j})|\alpha_{i,k,j}|}{\sum_{j=1}^{L+1}|\alpha_{i,k,j}|}$ where all of the $L+1$ channels' positions and recorded amplitudes contribute to the center of mass calculation.

The center of mass method is inexpensive to compute and has been shown to give informative location estimates for spikes in both real and synthetic data [34, 29, 17, 21]. Center of mass, however, suffers from two main drawbacks: First, since it is a convex method, location estimates must lie *inside* the bounds of chosen channels, negatively impacting location estimates for neurons outside of the MEA. Second, the center of mass is biased towards the central channel, leading to artificial separation of location estimates for spikes from the same neuron [34].

3 Method

In this section, we introduce a novel, model-based approach to spike localization. In the following subsections, we describe the model, the data augmentation procedure, and the inference methods.

3.1 Model

Our model uses the recorded amplitudes on each channel to determine the most likely source location of $s_{i,k}$. We assume that the peak signal from a spike decays exponentially with the distance from the source, r: $a\exp(br)$ where $a, b \in \mathbb{R}, r \in \mathbb{R}^+$. This assumption is well-motivated by experimentally recorded extracellular potential decay in both a salamander and mouse retina [39, 17], as well as a cat cortex [11]. This assumption has been further corroborated using realistic biophysical simulations [13].

We can utilize this exponential assumption to infer the source location of a spike, $s_{i,k}$, since localization is then equivalent to solving for $s_{i,k}$'s unknown parameters, $\theta_{s_{i,k}} \coloneqq \{a_{i,k}, b_{i,k}, x_{s_{i,k}}, y_{s_{i,k}}, z_{s_{i,k}}\}$ given the observed amplitudes, $\boldsymbol{\alpha}_{i,k}$. To allow for localization without knowing the identity of the firing neuron, we assume that each spike has individual exponential decay parameters, $a_{i,k}, b_{i,k}$, and individual source locations, $p_{s_{i,k}}$. We find, however, that fixing $b_{i,k}$ for all spikes to a constant that is equal to an empirical estimate from literature (decay length of $\sim 28\mu m$) works best across multiple probe geometries and noise levels, so we did not infer the value for $b_{i,k}$ in our final method. We will refer to the fixed decay rate as b and exclude it from the unknown parameters moving forward.

The generative process of the model is as follows,

$$\begin{aligned} &a_{i,k} \sim N(\mu_{a_{i,k}}, \sigma_a),\ x_{s_{i,k}} \sim N(\mu_{x_{s_{i,k}}}, \sigma_x),\ y_{s_{i,k}} \sim N(\mu_{y_{s_{i,k}}}, \sigma_y),\ z_{s_{i,k}} \sim N(\mu_{z_{s_{i,k}}}, \sigma_z) \\ &\hat{\boldsymbol{r}}_{i,k} = \|(x_{s_{i,k}}, y_{s_{i,k}}, z_{s_{i,k}}) - p_{\boldsymbol{c}}\|_2,\ \boldsymbol{\alpha}_{i,k} \sim \mathcal{N}(a_{i,k}\exp(b\hat{\boldsymbol{r}}_{i,k}), I) \end{aligned} \tag{1}$$

The amplitudes are drawn from an isotropic Gaussian distribution with a variance of one for computational simplicity since inferring the variance has a negligible effect on the localization result. We choose the prior distributions to be Gaussian since it is convenient to work with when using VAEs.

We found that inference, especially for a spike detected near the edge of the MEA, is sensitive to the mean of the prior distribution of $a_{i,k}$, therefore, we set $\mu_{a_{i,k}} = \lambda\alpha_{i,k,j_{min}}$ where $\alpha_{i,k,j_{min}}$ is the smallest negative amplitude peak of $s_{i,k}$. We choose this heuristic because the absolute value of

$\alpha_{i,k,j_{min}}$ will always be smaller than the absolute value of the amplitude of the spike at the source location, due to potential decay. Therefore, scaling $\alpha_{i,k,j_{min}}$ by λ gives a sensible value for $\mu_{a_{i,k}}$. We empirically choose $\lambda = 2$ for the final method after performing a grid search over $\lambda = \{1, 2, 3\}$. The parameter, σ_a, does not have a large affect on the inferred location so we set it to be approximately the standard deviation of the $\boldsymbol{\alpha}_{i,k,j_{min}}$. The location prior means, $\mu_{x_{s_{i,k}}}, \mu_{y_{s_{i,k}}}, \mu_{z_{s_{i,k}}}$, are set to the location of the minimum amplitude channel, $p_{c_{j_{min}}}$, for the given spike. The location prior standard deviations, $\sigma_x, \sigma_y, \sigma_z$, are set to large constant values to flatten out the distributions since we do not want the location estimate to be overly biased towards $p_{c_{j_{min}}}$.

3.2 Data Augmentation

In order to better differentiate between spikes detected near the center of the array (central spikes) and spikes detected near the edge of the array (edge spikes), we develop a novel data augmentation method. For the augmenation, we introduce virtual channels outside of the MEA, which have the same layout as the real, recording channels (see appendix C). Throughout this paper, we will refer to a virtual channel as an "unobserved" channel, c_{j_u}, and to a real channel on the MEA as an "observed" channel, c_{j_o}. We define the amplitude on an unobserved channel, c_{j_u}, to be zero since the unobserved channels do not actually record any signals. The amplitude for an observed channel, c_{j_o}, is defined as $\alpha_{i,k,j_o} \coloneqq \min_t w_{i,k,j_o}$ as before.

To construct the augmented dataset for a spike, $s_{i,k}$, we take the set of L channels that lie within a bounding box of width W centered on the *observed* channel with the minimum recorded amplitude, $c_{j_{o_{min}}}$. For edge spikes where $c_{j_{o_{min}}}$ lies near the bounds of the MEA, their augmented data contain more unobserved channels than does the augmented data of central spikes. The amount of unobserved channels in the augmented data then helps to discriminate edge spikes from central spikes. To define the augmented dataset, we must first define the indicator function, $1_o : \alpha \to \{0, 1\}$:

$$1_o(\alpha) = \begin{cases} 1, & \text{if } \alpha \text{ is from an observed channel,} \\ 0, & \text{if } \alpha \text{ is from an unobserved channel.} \end{cases}$$

where α is an amplitude from any channel, observed or unobserved. Using this function, we define our newly augmented observed data for $s_{i,k}$ as,

$$\boldsymbol{\beta}_{i,k} \coloneqq \{(\alpha_{i,k,j}, 1_o(\alpha_{i,k,j})\}_{j=1}^{L} \tag{2}$$

So for a single spike, we construct a $L \times 2$ dimensional vector that contains amplitudes from L channels and indices indicating which amplitudes came from observed or unobserved channels.

3.3 Inference

Now that we have defined the generative process and data augmentation procedure, we would like to compute the posterior distribution for the unknown parameters of a spike, $s_{i,k}$,

$$p(a_{i,k}, x_{s_{i,k}}, y_{s_{i,k}}, z_{s_{i,k}} | \boldsymbol{\beta}_{i,k}) \tag{3}$$

given the augmented dataset, $\boldsymbol{\beta}_{i,k}$. To infer the posterior distribution for each spike, we tried two methods of Bayesian inference: MCMC sampling and amortized variational inference.

MCMC sampling We use MCMC to assess the validity and applicability of our model to extracellular data. We implement our model in the probabilistic modeling framework Turing [10] in Julia. We run Hamiltonian Monte Carlo (HMC) [30] for 10,000 iterations with a step size of 0.01 and a step number of 10. We use the posterior means of the location distributions as the estimated location.[1]

Despite the ease use of probabilistic programming and asymptotically guaranteed inference quality of MCMC methods, the scalability of MCMC methods to large-scale datasets is limited. This leads us to implement our model as a VAE and to perform amortized variational inference for our final method.

Amortized variational inference To speed up inference of the spike parameters, we construct a VAE and use amortized variational inference to estimate posterior distributions for each spike. In variational inference, instead of sampling from the target intractable posterior distribution of interest,

[1] The code for our MCMC implementation is provided in Appendix H.

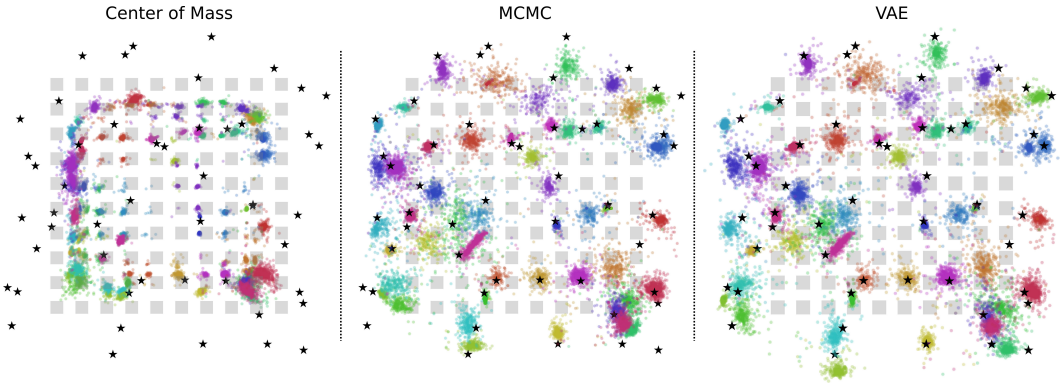


Figure 1: *Estimated spike locations for the different methods on a 10muV recording.* Center of mass estimates (left) are calculated using 16 observed amplitudes. The MCMC estimated locations (middle) used 9-25 observed amplitudes for inference, and the VAE model (right) was trained on 9-25 observed amplitudes and a 10 amplitude jitter (amplitude jitter is described in 3.3).

we construct a variational distribution that is tractable and minimize the Kullback–Leibler (KL) divergence between the variational posterior and the true posterior. Minimizing the KL divergence is equivalent to maximizing the evidence lower bound (ELBO) for the log marginal likelihood of the data. In VAEs, the parameters of the variational posterior are not optimized directly, but are, instead, computed by an inference network.

We define our variational posterior for x, y, z as a multivariate Normal with diagonal covariance where the mean and diagonal of the covariance matrix are computed by an inference network

$$q_\Phi(x, y, z) = \mathcal{N}(\boldsymbol{\mu}_{\phi_1}(f_{\phi_0}(\upsilon_{i,k})), \boldsymbol{\sigma}^2_{\phi_2}(f_{\phi_0}(\upsilon_{i,k}))) \tag{4}$$

The inference network is implemented as a feed-forward, deep neural network parameterized by $\Phi = \{\phi_0, \phi_1, \phi_2\}$. As one can see, the variational parameters are a function of the input $\boldsymbol{\upsilon}$.

When using an inference network, the input can be any part of the dataset so for our method, we use, $\boldsymbol{\upsilon_{i,k}}$, as the input for each spike, $s_{i,k}$, which is defined as follows:

$$\boldsymbol{\upsilon}_{i,k} \coloneqq \{(w_{i,k,j}, 1_o(\alpha_{i,k,j})\}_{j=1}^{L} \tag{5}$$

where $w_{i,k,j}$ is the waveform detected on the jth channel (defined in Section 2). Similar to our previous augmentation, the waveform for an unobserved channel is set to be all zeros. We choose to input the waveforms rather than the amplitudes because, empirically, it encourages the inferred location estimates for spikes from the same neuron to be better localized to the same region of the MEA. For both the real and simulated datasets, we used $\sim$2 ms of readings for each waveform.

The decoder for our method reconstructs the *amplitudes* from the observed data rather than the waveforms. Since we assume an exponential decay for the amplitudes, the decoder is a simple Gaussian likelihood function, where given the Euclidean distance vector $\hat{\boldsymbol{r}_{i,k}}$, computed by samples from the variational posterior, the decoder reconstructs the mean value of the observed amplitudes with a fixed variance. The decoder is parameterized by the exponential parameters of the given spike, $s_{i,k}$, so it reconstructs the amplitudes of the augmented data, $\boldsymbol{\beta}^{(0)}_{i,k}$, with the following expression: $\hat{\boldsymbol{\beta}}^{(0)}_{i,k} \coloneqq a_{i,k} \exp(b\hat{\boldsymbol{r}}_{i,k}) \times \beta^1_{i,k}$ where $\hat{\boldsymbol{\beta}}^{(0)}_{i,k}$ is the reconstructed observed amplitudes. By multiplying the reconstructed amplitude vector by $\beta^1_{i,k}$ which consists of either zeros or ones (see Eq. 5), the unobserved channels will be reconstructed with amplitudes of zero and the observed channels will be reconstructed with the exponential function. For our VAE, instead of estimating the distribution of $a_{i,k}$, we directly optimize $a_{i,k}$ when maximizing the lower bound. We set the initial value of $a_{i,k}$ to the mean of the prior. Thus, $a_{i,k}$ can be read as a parameter of the decoder.

Given our inference network and decoder, the ELBO we maximize for each spike, $s_{i,k}$, is given by,

$$\log p(\boldsymbol{\beta}_{i,k}; a_{i,k}) \geq -\text{KL}\left[q_\Phi(x, y, z) \,\|\, p_x p_y p_z\right] + \mathbb{E}_{q_\Phi}\left[\sum_{l=1}^{L} \mathcal{N}(\beta^0_{i,k,l} | a_{i,k} \exp(b\hat{\boldsymbol{r}}_{i,k}), I)\beta^1_{i,k,l}\right]$$

Method	Observed Channels	2D Avg. Spike Distance from Soma (μm)		
		10 μV	20 μV	30 μV
COM	4	15.84±10.08	16.46±10.39	17.18±10.97
COM	9	18.05±11.42	18.59±11.67	19.22±12.1
COM	16	20.94±13.09	21.54±13.46	22.17±13.94
COM	25	23.44±14.81	24.31±15.43	25.18±15.98
MCMC	9-25	9.87±8.64	11.30±9.85	13.31±11.67
VAE - 0μV	4-9	9.21±8.00	10.40±8.97	12.05±10.35
VAE - 10μV	4-9	**8.79±7.49**	**9.79±8.31**	**11.18±9.56**
VAE - 0μV	9-25	8.94±7.91	10.48±9.334	12.43±11.223
VAE - 10μV	9-25	9.12±7.83	10.41±9.07	12.27±10.78

Table 1: Results of the localization methods on the three simulated datasets from the square MEA with noise levels ranging from 10μV-30μV. The number next to the VAE methods in the first column is the amount of amplitude jitter that was used for the method (as described in 3.3).

where KL is the KL-divergence. The location priors, p_x, p_y, p_z, are normally distributed as described in 3.1, with means of zero and variances of 80. For more information about the architecture and training, see Appendix F.

Stabilized Location Estimation In this model, the channel on which the input is centered, $c_{j_{o_{min}}}$, can bias the estimate of the spike location, in particular when amplitudes are small. To reduce this bias, we can average the inferred locations of multiple inputs for the same spike *centered on different channels* after training. To this end, we introduce a hyperparameter, amplitude jitter, where for each spike, $s_{i,k}$, we create multiple inputs centered on channels with peak amplitudes within the set range of the maximum amplitude, $\alpha_{i,k,j}$. The inference network is trained normally, but during inference, the mean location posterior estimates of all inputs that belong to the same spike are averaged. We use two values for the amplitude jitter in our experiments: $0\mu V$ and $10\mu V$. When amplitude jitter is set to $0\mu V$, no averaging is performed.

4 Experiments

4.1 Datasets

We simulate biophysically realistic ground truth extracellular recordings to test our model against a variety of real-life complexities. The simulations were run using the `MEArec`[2] package which includes 13 neuron models from the layer 5 of a juvenile rat somatosensory cortex from the neocortical microcircuit collaboration portal [35]. We simulate three recordings with different noise levels (ranging from $10\mu V$ to $30\mu V$) for two probe geometries: a 10x10 channel square MEA with a 15 μm inter-channel distance and 64 channels from a Neuropixels probe ($\sim$25-40 μm inter-channel distance). Our simulations contain 40 excitatory cells and 10 inhibitory cells with random morphological subtypes, randomly distributed and rotated in 3D space around the probe (20 μm minimum distance between somas). Each dataset has around 20,000 spikes in total (60s). For more details on the simulation and noise model, see Appendix G.

For real datasets, we use public data from a Neuropixels probe [25] and from a mouse retina recorded with the BioCam4096 platform [19]. The two datasets have 6 million and 2.2 million spikes, respectively. Spike detection and sorting (with our location estimates) is done using the HerdingSpikes2 software [17].

4.2 Evaluation Metrics

To evaluate the localization methods on the simulated data, we first detect the spikes in each recording. To avoid biasing the results by our choice of detection algorithm, we assume perfect detection, extracting waveforms from channels near each spiking neuron. We then generate location estimates for all the datasets using each method.

[2]https://github.com/alejoe91/MEArec, ref. [4]

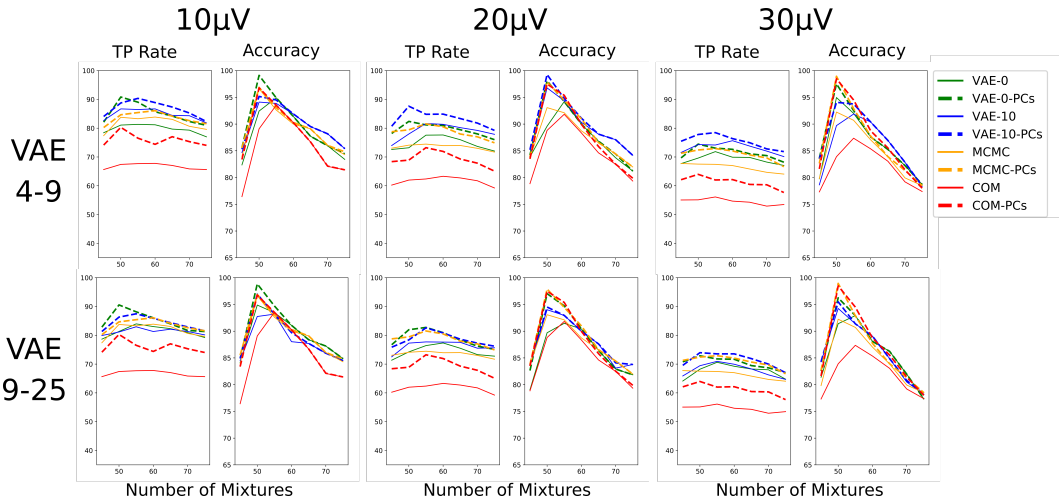


Figure 2: *Spike Sorting Performance on square MEA*. We compare the sorting performance of all localization methods with and without principal components across all noise levels. For the VAE, we include the results with 0μV and 10μV amplitude jitter and with different amounts of observed channels (4-9 and 9-25). For COM, we plot the highest sorting performance (16 observed channels). The test data set had 50 neurons.

In order to evaluate the accuracy of each method, we compute the 2D Euclidean distance between the estimated spike locations and the associated firing neurons. We report the mean and standard deviation of the localization error for all spikes in each recording.

To evaluate the effectiveness of each method for spike sorting, we perform two sorting experiments.[3] First, we cluster the 2D location estimates using Gaussian mixture models (GMMs). The GMMs are fit with spherical covariances ranging from 45 to 75 mixture components with a step size of 5. We report the true positive rate and accuracy for each number of mixture components when matched back to ground truth. Second, we use GMMs to cluster the 2D location estimates combined with two principal components from each spike. We report the true positive rate and accuracy for each number of mixture components as before. Combining location estimates and waveform information was introduced in Hilgen (2017), where the principal components were whitened and then scaled with a hyperparameter, α. To remove any bias from choosing an α value in our experiments, we conduct a grid search over $\alpha = \{4, 6, 8, 10\}$ and report the best metric scores for each method.

To test the generalization performance of the method, we train a VAE on an extracellular dataset and then try to infer the spike locations in another dataset where the neuron locations are different, but all other aspects are kept the same (10μV, square MEA). The performance is compared to that of a model trained directly on the second dataset.

Overall, we use the first experiment to show how useful each method is purely as a localization tool. We use the second experiment to show how useful the location estimates are for spike sorting immediately after localizing. We use the third experiment to show how much the performance *can* improve given extra waveform information. Finally, we use the fourth experiment to highlight how our method potentially could be used across datasets without retraining.

4.3 Results

Table 1 reports the localization accuracy of different localization methods on the square MEA over three noise levels. The model-based methods far outperform center of mass with any number of observed channels. As expected, introducing amplitude jitter helps lower the mean and standard deviation of the location spike distance. Using a small width of 20 μm when constructing the augmented data (4-9 observed channels) has the highest performance for the square MEA.

The location estimates for the square MEA are visualized in Figure 1. In this figure, recording channels are shown in grey and true soma locations are shown in black. The estimated individual

[3]For our sorting analysis, we use SpikeInterface (https://github.com/SpikeInterface).

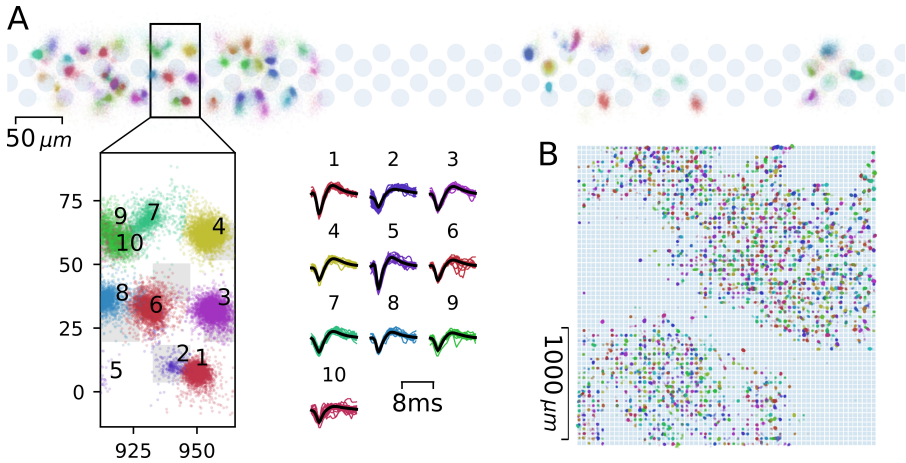


Figure 3: *Estimated spike locations for two real recordings.* A, Analysis of a one hour recording from an awake, head-fixed mouse with a Neuropixels probe. Spikes were detected using the HS2 package [17], their locations estimated using the VAE model, and clustered with mean shift, together with the first two principal components obtained from the waveforms. Shown are a large section of the probe, a magnification and corresponding spike waveforms from the clustered units. B, The same analysis performed on a recording from a mouse retina with a BioCam array from ref [19].

spike locations are colored according to their associated firing neuron identity. The center of mass location estimates suffers both from artificial splitting of location estimates and poor performance on neurons outside the array, two areas in which the model-based approaches excel. The MCMC and VAE methods have very similar location estimates, highlighting the success of our variational inference in approximating the true posterior.

In Figure 2, spike sorting performance on the square MEA is visualized for all localization methods. Here we only show the sorting results of the center of mass with the highest performance. Overall, the model-based approaches have significantly higher true positive rates than the best center of mass results and have comparable accuracies across all noise levels. This highlights how model-based location estimates provide a much more discriminatory feature set than the location estimates from the center of mass approaches. Interestingly, the VAE localization estimates had higher sorting performance than the MCMC location estimates. We attribute this performance increase to the local coupling of parameters in the inference network that have a smoothing effect on the location estimates. See Appendix A for the same sorting analysis on the simulated Neuropixels recordings.

Figure 3 visualizes our method as applied to two real datasets. We evaluate our approach on extracellular recordings without ground truth as current extracellular ground truth is limited to a few labeled neurons [42, 14, 16, 31, 40]. To show that the units we find likely correspond to individual neurons, we visualize waveforms from a local grouping of sorted units on the Neuropixels probe.

Our method generalizes well to other recordings with different neuron locations as the localization accuracy and sorting performance drop only slightly in comparison to the VAE that is trained directly on the other recording. Our method also still outperforms center of mass on the new dataset. We show as well that the inference time for the VAE is much faster than that of MCMC, highlighting the excellent scalability of our method. The inference speed of the VAE allows for localization of one million spikes in approximately 37 seconds, enabling real-time analysis of large-scale extracellular datasets. Since we are limited for space, see Appendices D and E for these analyses.

5 Conclusion

Here we introduce a novel method for spike localization that significantly improves localization accuracy and sorting performance over preexisting baselines while remaining scalable to the large volumes of data generated by MEAs. Scalability is particularly relevant for recordings from thousands of channels, where a single experiment may yield in the order of 100 million spikes to be analyzed.

We validate the accuracy of our model assumptions and inference scheme using biophysically realistic ground truth simulated recordings that capture much of the variability seen in real recordings. Despite the realism of our simulated recordings, there are some factors that we did not account for, including: bursting cells with event amplitude fluctuations (although this should not affect the spatial amplitude profile), electrode drift, and realistic intrinsic variability of recorded spike waveforms. As these additional factors are difficult to model, future analysis of real recordings will help to understand possible limitations of the method. Along with the simulated dataset, we also apply our method to two real datasets, providing evidence that it can be used on large-scale extracellular recordings.

Other work indicates there is scope for extending the model to combine event detection and localization [24], and to distinguish between different morphological neuron types based on their activity footprint on the array [3]. Our work is thus a first step towards using highly efficient amortized variational inference for the unsupervised analysis of complex electrophysiological recordings.

References

[1] *Ballini Marco, Muller Jan, Livi Paolo, Chen Yihui, Frey Urs, Stettler Alexander, Shadmani Amir, Viswam Vijay, Jones Ian Lloyd, Jackel David, Radivojevic Milos, Lewandowska Marta K., Gong Wei, Fiscella Michele, Bakkum Douglas J., Heer Flavio, Hierlemann Andreas*. A 1024-channel CMOS microelectrode array with 26,400 electrodes for recording and stimulation of electrogenic cells in vitro // IEEE Journal of Solid-State Circuits. 2014. 49, 11. 2705–2719.

[2] *Berdondini L, Wal P D van der, Guenat O, Rooij N F de, Koudelka-Hep M, Seitz P, Kaufmann R, Metzler P, Blanc N, Rohr S*. High-density electrode array for imaging in vitro electrophysiological activity. // Biosensors & Bioelectronics. jul 2005. 21, 1. 167–74.

[3] *Buccino Alessio P, Ness Torbjørn V, Einevoll Gaute T, Cauwenberghs Gert, Häfliger Philipp D*. Localizing neuronal somata from Multi-Electrode Array in-vivo recordings using deep learning // Engineering in Medicine and Biology Society (EMBC), 2017 39th Annual International Conference of the IEEE. 2017. 974–977.

[4] *Buccino Alessio Paolo, Kordovan Michael, Ness Torbjørn V Bækø, Merkt Benjamin, Häfliger Philipp Dominik, Fyhn Marianne, Cauwenberghs Gert, Rotter Stefan, Einevoll Gaute T*. Combining biophysical modeling and deep learning for multi-electrode array neuron localization and classification // Journal of neurophysiology. 2018.

[5] *Chung Jason E, Magland Jeremy F, Barnett Alex H, Tolosa Vanessa M, Tooker Angela C, Lee Kye Y, Shah Kedar G, Felix Sarah H, Frank Loren M, Greengard Leslie F*. A fully automated approach to spike sorting // Neuron. 2017. 95, 6. 1381–1394.

[6] *Dayan Peter, Hinton Geoffrey E, Neal Radford M, Zemel Richard S*. The helmholtz machine // Neural computation. 1995. 7, 5. 889–904.

[7] *Dimitriadis George, Neto Joana P, Aarts Arno, Alexandru Andrei, Ballini Marco, Battaglia Francesco, Calcaterra Lorenza, David Francois, Fiath Richard, Frazao Joao, others* . Why not record from every channel with a CMOS scanning probe? // bioRxiv. 2018. 275818.

[8] *Eversmann Björn, Jenkner Martin, Hofmann Franz, Paulus Christian, Brederlow Ralf, Holzapfl Birgit, Fromherz Peter, Merz Matthias, Brenner Markus, Schreiter Matthias, Gabl Reinhard, Plehnert Kurt, Steinhauser Michael, Eckstein Gerald, Schmitt-landsiedel Doris, Thewes Roland*. A 128 128 CMOS Biosensor Array for Extracellular Recording of Neural Activity // IEEE Journal of Solid-State Circuits. 2003. 38, 12. 2306–2317.

[9] *Frey Urs, Sedivy Jan, Heer Flavio, Pedron Rene, Ballini Marco, Mueller Jan, Bakkum Douglas, Hafizovic Sadik, Faraci Francesca D., Greve Frauke, Kirstein Kay Uwe, Hierlemann Andreas*. Switch-matrix-based high-density microelectrode array in CMOS technology // IEEE Journal of Solid-State Circuits. 2010. 45, 2. 467–482.

[10] *Ge Hong, Xu Kai, Ghahramani Zoubin*. Turing: Composable inference for probabilistic programming // International Conference on Artificial Intelligence and Statistics. 2018. 1682–1690.

[11] *Gray Charles M, Maldonado Pedro E, Wilson Mathew, McNaughton Bruce*. Tetrodes markedly improve the reliability and yield of multiple single-unit isolation from multi-unit recordings in cat striate cortex // Journal of Neuroscience Methods. 1995. 63, 1-2. 43–54.

[12] *Hagen Espen, Næss Solveig, Ness Torbjørn V, Einevoll Gaute T*. Multimodal Modeling of Neural Network Activity: Computing LFP, ECoG, EEG, and MEG Signals With LFPy 2.0 // Frontiers in Neuroinformatics. 2018. 12.

[13] *Hagen Espen, Ness Torbjørn V., Khosrowshahi Amir, Sørensen Christina, Fyhn Marianne, Hafting Torkel, Franke Felix, Einevoll Gaute T.* ViSAPy: A Python tool for biophysics-based generation of virtual spiking activity for evaluation of spike-sorting algorithms // Journal of Neuroscience Methods. 2015. 245. 182–204.

[14] *Harris Kenneth D, Henze Darrell A, Csicsvari J, Hirase H, Buzsáki G*. Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements. // Journal of Neurophysiololgy. 2000. 84, 1. 401–414.

[15] *Hennig Matthias H, Hurwitz Cole, Sorbaro Martino*. Scaling Spike Detection and Sorting for Next Generation Electrophysiology // arXiv preprint arXiv:1809.01051. 2018.

[16] *Henze Darrell A, Borhegyi Zsolt, Csicsvari Jozsef, Mamiya Akira, Harris Kenneth D, Buzsaki Gyorgy*. Intracellular features predicted by extracellular recordings in the hippocampus in vivo // Journal of neurophysiology. 2000. 84, 1. 390–400.

[17] *Hilgen Gerrit, Sorbaro Martino, Pirmoradian Sahar, Muthmann Jens-Oliver, Kepiro Ibolya Edit, Ullo Simona, Ramirez Cesar Juarez, Encinas Albert Puente, Maccione Alessandro, Berdondini Luca, others* . Unsupervised spike sorting for large-scale, high-density multielectrode arrays // Cell Reports. 2017. 18, 10. 2521–2532.

[18] *Hines Michael L, Carnevale Nicholas T*. The NEURON simulation environment // Neural Computation. 1997. 9, 6. 1179–1209.

[19] *Jouty Jonathan, Hilgen Gerrit, Sernagor Evelyne, Hennig Matthias H*. Non-parametric physiological classification of retinal ganglion cells in the mouse retina // Frontiers in Cellular Neuroscience. 2018. 12. 481.

[20] *Jun James J, Steinmetz Nicholas A, Siegle Joshua H, Denman Daniel J, Bauza Marius, Barbarits Brian, Lee Albert K, Anastassiou Costas A, Andrei Alexandru, Aydın Çağatay, others* . Fully integrated silicon probes for high-density recording of neural activity // Nature. 2017. 551, 7679. 232.

[21] *Jun James Jaeyoon, Mitelut Catalin, Lai Chongxi, Gratiy Sergey, Anastassiou Costas, Harris Timothy D*. Real-time spike sorting platform for high-density extracellular probes with ground-truth validation and drift correction // bioRxiv. 2017. 101030.

[22] *Kelly Ryan C, Smith Matthew A, Samonds Jason M, Kohn Adam, Bonds AB, Movshon J Anthony, Lee Tai Sing*. Comparison of recordings from microelectrode arrays and single electrodes in the visual cortex // Journal of Neuroscience. 2007. 27, 2. 261–264.

[23] *Kingma Diederik P, Welling Max*. Auto-encoding variational bayes // arXiv preprint arXiv:1312.6114. 2013.

[24] *Lee Jin Hyung, Carlson David E, Razaghi Hooshmand Shokri, Yao Weichi, Goetz Georges A, Hagen Espen, Batty Eleanor, Chichilnisky EJ, Einevoll Gaute T, Paninski Liam*. YASS: Yet Another Spike Sorter // Advances in Neural Information Processing Systems. 2017. 4005–4015.

[25] *Lopez Carolina Mora, Mitra Srinjoy, Putzeys Jan, Raducanu Bogdan, Ballini Marco, Andrei Alexandru, Severi Simone, Welkenhuysen Marleen, Van Hoof Chris, Musa Silke, others* . 22.7 A 966-electrode neural probe with 384 configurable channels in 0.13 μm SOI CMOS // Solid-State Circuits Conference (ISSCC), 2016 IEEE International. 2016. 392–393.

[26] *Mechler Ferenc, Victor Jonathan D, Ohiorhenuan Ifije E, Schmid Anita M, Hu Qin*. Three-dimensional localization of neurons in cortical tetrode recordings // American Journal of Physiology-Heart and Circulatory Physiology. 2011.

[27] *Miller Earl K, Wilson Matthew A*. All my circuits: using multiple electrodes to understand functioning neural networks // Neuron. 2008. 60, 3. 483–488.

[28] *Müller Jan, Ballini Marco, Livi Paolo, Chen Yihui, Radivojevic Milos, Shadmani Amir, Viswam Vijay, Jones Ian L, Fiscella Michele, Diggelmann Roland, Stettler Alexander, Frey Urs, Bakkum Douglas J, Hierlemann Andreas, Muller Jan, Ballini Marco, Livi Paolo, Chen Yihui, Radivojevic Milos, Shadmani Amir, Viswam Vijay, Jones Ian L, Fiscella Michele, Diggelmann Roland, Stettler Alexander, Frey Urs, Bakkum Douglas J, Hierlemann Andreas*. High-resolution CMOS MEA platform to study neurons at subcellular, cellular, and network levels // Lab on a Chip. 2015. 15, 13. 2767–2780.

[29] *Muthmann Jens-Oliver, Amin Hayder, Sernagor Evelyne, Maccione Alessandro, Panas Dagmara, Berdondini Luca, Bhalla Upinder S, Hennig Matthias H*. Spike Detection for Large Neural Populations Using High Density Multielectrode Arrays // Frontiers in Neuroinformatics. dec 2015. 9, December. 1–21.

[30] *Neal Radford M, others* . MCMC using Hamiltonian dynamics // Handbook of markov chain monte carlo. 2011. 2, 11. 2.

[31] *Neto Joana P, Lopes Gonçalo, Frazão João, Nogueira Joana, Lacerda Pedro, Baião Pedro, Aarts Arno, Andrei Alexandru, Musa Silke, Fortunato Elvira, others* . Validating silicon polytrodes with paired juxtacellular recordings: method and dataset // Journal of Neurophysiology. 2016. 116, 2. 892–903.

[32] *Obien Marie Engelene J, Deligkaris Kosmas, Bullmann Torsten, Bakkum Douglas J., Frey Urs*. Revealing neuronal function through microelectrode array recordings // Frontiers in Neuroscience. 2015. 9, JAN. 423.

[33] *Pachitariu Marius, Steinmetz Nicholas A, Kadir Shabnam N, Carandini Matteo, Harris Kenneth D*. Fast and accurate spike sorting of high-channel count probes with KiloSort // Advances in Neural Information Processing Systems. 2016. 4448–4456.

[34] *Prentice Jason S, Homann Jan, Simmons Kristina D, Tkačik Gašper, Balasubramanian Vijay, Nelson Philip C*. Fast, scalable, Bayesian spike identification for multi-electrode arrays. // PloS One. jan 2011. 6, 7. e19884.

[35] *Ramaswamy Srikanth, Courcol Jean-Denis, Abdellah Marwan, Adaszewski Stanislaw R, Antille Nicolas, Arsever Selim, Atenekeng Guy, Bilgili Ahmet, Brukau Yury, Chalimourda Athanassia, others* . The neocortical microcircuit collaboration portal: a resource for rat somatosensory cortex // Frontiers in Neural Circuits. 2015. 9. 44.

[36] *Rey Hernan Gonzalo, Pedreira Carlos, Quian Quiroga Rodrigo*. Past, present and future of spike sorting techniques // Brain Research Bulletin. 2015. 119. 106–117.

[37] *Rezende Danilo Jimenez, Mohamed Shakir, Wierstra Daan*. Stochastic backpropagation and approximate inference in deep generative models // arXiv preprint arXiv:1401.4082. 2014.

[38] *Ruz Isabel Delgado, Schultz Simon R*. Localising and classifying neurons from high density MEA recordings // Journal of neuroscience methods. 2014. 233. 115–128.

[39] *Segev Ronen, Goodhouse Joe, Puchalla Jason, Berry II Michael J*. Recording spikes from a large fraction of the ganglion cells in a retinal patch // Nature Neuroscience. 2004. 7, 10. 1155.

[40] *Yger Pierre, Spampinato Giulia LB, Esposito Elric, Lefebvre Baptiste, Deny Stéphane, Gardella Christophe, Stimberg Marcel, Jetter Florian, Zeck Guenther, Picaud Serge, others* . A spike sorting toolbox for up to thousands of electrodes validated with ground truth recordings in vitro and in vivo // eLife. 2018. 7. e34518.

[41] *Yuan X, Kim S, Juyon J, D'Urbino M, Bullmann T, Chen Y, Stettler Alexander, Hierlemann Andreas, Frey Urs*. A microelectrode array with 8,640 electrodes enabling simultaneous full-frame readout at 6.5 kfps and 112-channel switch-matrix readout at 20 kS/s // VLSI Circuits (VLSI-Circuits), 2016 IEEE Symposium on. 2016. 1–2.

[42] *Zanoci Cristian, Dehghani Nima, Tegmark Max*. Ensemble inhibition and excitation in the human cortex: An Ising-model analysis with uncertainties // Physical Review E. 2019. 99, 3. 032408.

Appendix

A Neuropixels Results

Table 2: Results of the localization methods on three simulated datasets from a 64 channel Neuropixels probe with noise levels ranging from 10μV-30μV. The number next to the VAE methods in the first column is the amount of amplitude jitter that was used for the method (as described in 3.3).

Method	Observed Channels	2D Avg. Spike Distance from Soma (μm)		
		10 μV	20 μV	30 μV
COM	4	23.85±12.95	25.16±14.21	26.66±15.6
COM	7	22.81±14.04	24.36±15.25	26.11±16.63
COM	12	26.33±15.55	28.26±16.66	30.1±17.81
COM	14	27.83±16.26	30.08±17.48	32.03±18.57
MCMC	8-14	14.28± 12.68	16.80±15.45	19.74±18.30
VAE - 0	3-6	14.25±12.88	15.74±14.88	18.44±17.68
VAE - 10	3-6	13.10±11.04	**15.20±13.66**	**17.68±16.38**
VAE - 0	8-14	13.31±12.46	15.63±15.51	18.49±18.89
VAE - 10	8-14	**12.91±11.41**	15.38±14.35	18.14±17.55

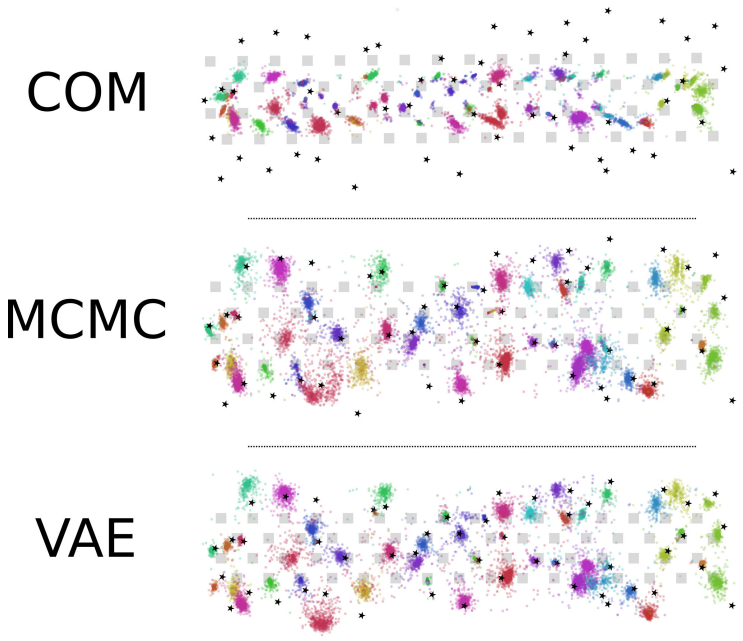


Figure 4: *Estimated spike locations for the different methods on a 10μV recording.* Center of mass estimates (top) are calculated using 7 channels. The MCMC estimated locations (middle) used 8-14 channels of observed amplitudes for inference, and the VAE model (bottom) was trained on 8-14 channels surrounding each spike and 0 amplitude jitter (see 3.3 for amplitude jitter explanation).

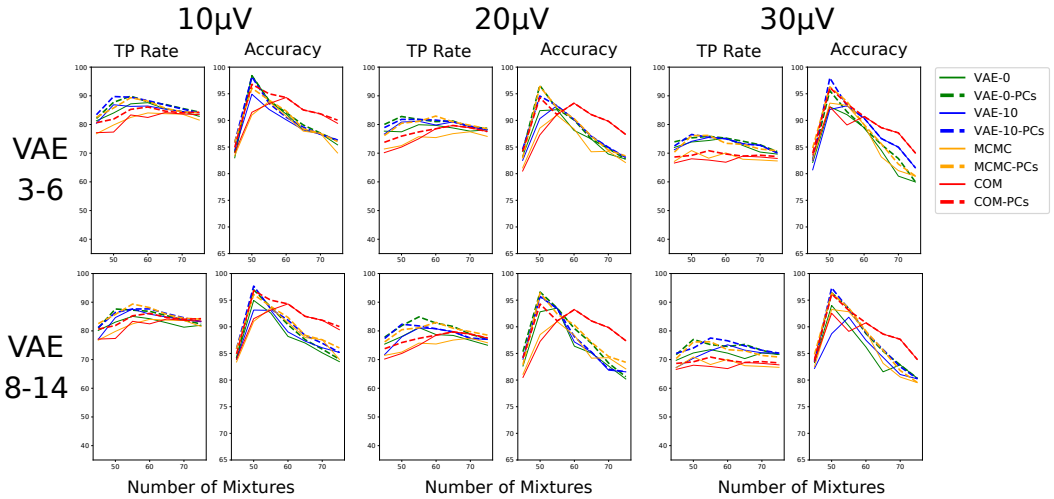


Figure 5: *Spike Sorting Performance on Neuropixels.* We compare the sorting performance of all localization methods with and without principal components across all noise levels. For the VAE, we include the results with and without amplitude jitter and with different amounts of real channels. For COM, we plot the highest sorting performance which was 4 observed channels.

B Effect of Noise on VAE

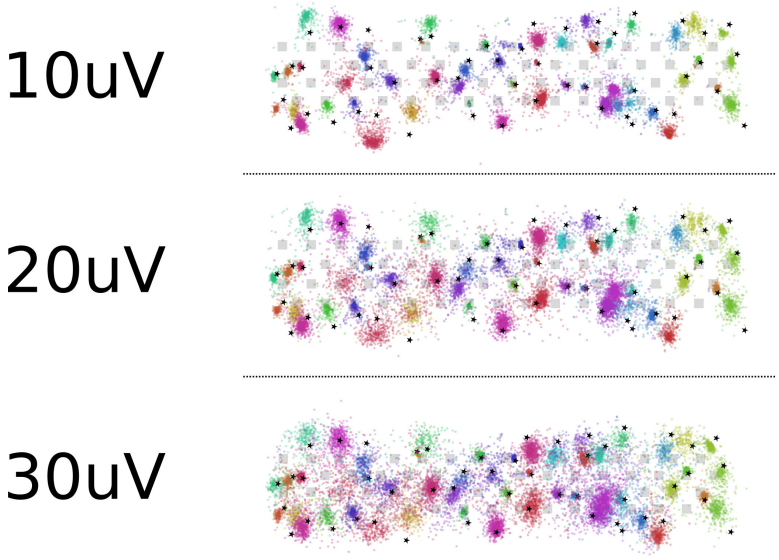


Figure 6: *Effect of noise on location inference for the VAE on the Neuropixels probe.* We vary the noise levels for the recording from 10μV, 20μV, and 30μV. Increasing the noise also increases the number of outliers in and the spread of the location estimates.

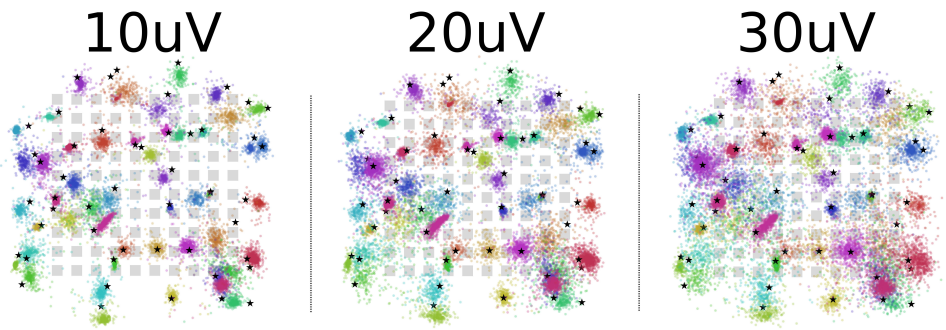


Figure 7: *Effect of noise on location inference for the VAE on the square MEA*. We vary the noise levels for the recording from 10μV, 20μV, and 30μV. Increasing the noise also increases the number of outliers in and the spread of the location estimates.

C Data Augmentation

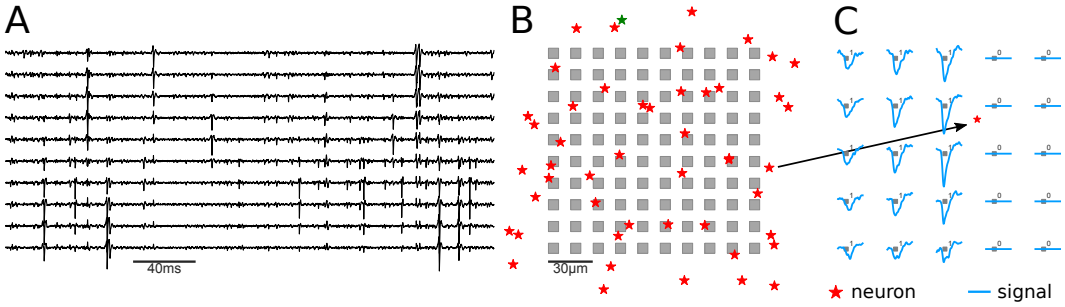


Figure 8: *The simulated recording set-up and example data.* A, Example electrical traces from the MEA with recorded action potentials (spikes, negative deflections). B, The 2D layout of the simulated recording. Recording channels are indicated in grey, and the true locations of the simulated neurons in red. The traces in part A are taken from the first column of the array. Note each spike is visible in multiple channels, with a characteristic spatial decay. C, Illustration of the data augmentation procedure in cases where the spikes are detected on channels near the array boundary. A set of virtual channels is introduced, which are incapable of recording any signal, but would report non-zero amplitudes if they were present on the MEA.

D Generalization Performance

Table 3: Location results for the generalization performance of a VAE trained on one 10μV, square MEA dataset and tested on another 10μV, square MEA dataset. We compare the results of this VAE to another VAE that is trained directly on the second dataset to quantify the drop in performance when generalizing between datasets. We also compare to the center of mass baselines.

Method	Observed Channels	2D Avg. Spike Distance from Soma (microns)
COM	4	16.53 $\pm$ 10.83
COM	9	18.25 $\pm$ 13.0
COM	16	20.41 $\pm$ 14.57
COM	25	22.73 $\pm$ 16.32
VAE - 0 - Trained	9-25	11.57 $\pm$ 9.88
VAE - 0 - Inferred	9-25	13.73 $\pm$ 8.01

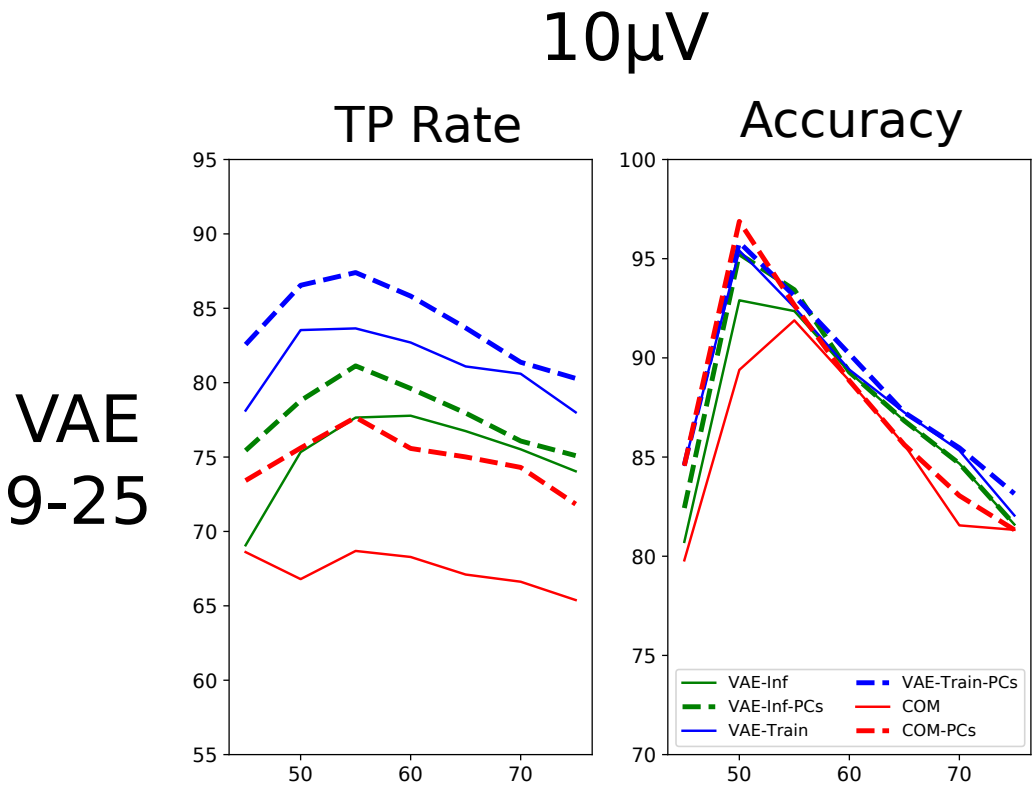


Figure 9: *Spike Sorting Performance Generalization.* We compare the sorting performance of the location estimates from a VAE that is trained on one 10μV, square MEA dataset and then used to infer locations for another 10μV, square MEA dataset. We compare the sorting results of this VAE to another set of location estimates from another VAE that is trained directly on the second dataset. We compare both methods to the best center of mass baseline (16 observed channels). This result highlights how a trained network can generalize to a new dataset.

E Inference Time

Table 4: Results for the inference time of the VAE versus HMC sampling on the dataset. We ran HMC for 10,000 iterations. The VAE was run on a TITAN X GPU.

Method	Per Spike Inference Time (s)	Dataset Inference Time (s)
MCMC	0.343	6669.0
VAE	0.000037	0.722

F Architecture and Training Details

We set the inference network to be 2 layers deep with ReLU nonlinearities. The hidden unit sizes in the inference network are set to be [500, 250]. We include batchnorm layers throughout the encoder to improve training and generalization.

We train the VAE with three different learning rates, $\{.0003, .001, .003\}$, and choose the learning rate that has the highest performance, although this parameter did not have a large effect on the results.

To ensure convergence for the simulated data, we train the network for 400 epochs on the entire dataset. For the real datasets, we train the network on a subset of the detected spikes ($\sim$100,000 spikes) and then we infer the rest of the locations.

G Simulated Data

To generate the extracellular recordings, we simulate the multi-compartment neuron models using NEURON [18] and use the transmembrane currents to compute extracellular action potentials (EAP) with LFPy [12]. EAPs are then combined with randomly generated spike trains to generate recordings. Finally, noise is added and the entire recording is filtered using a 3rd order Butterworth filter (0.3, 6 kHz).

For the noise model, we simulate templates for 300 neurons that are far away from the recording area. These small action potentials make up the background noise of the recording and have noise levels ranging from $10\mu V$ to $30\mu V$ standard deviation for the simulated datasets. We choose this noise model because it best captures the frequency and challenges of background noise in real extracellular recordings.

For each of the three recordings on one probe geometry, we fix the neuron locations to assess the effect of noise on the location estimates for each neuron.

H MCMC Turing Code

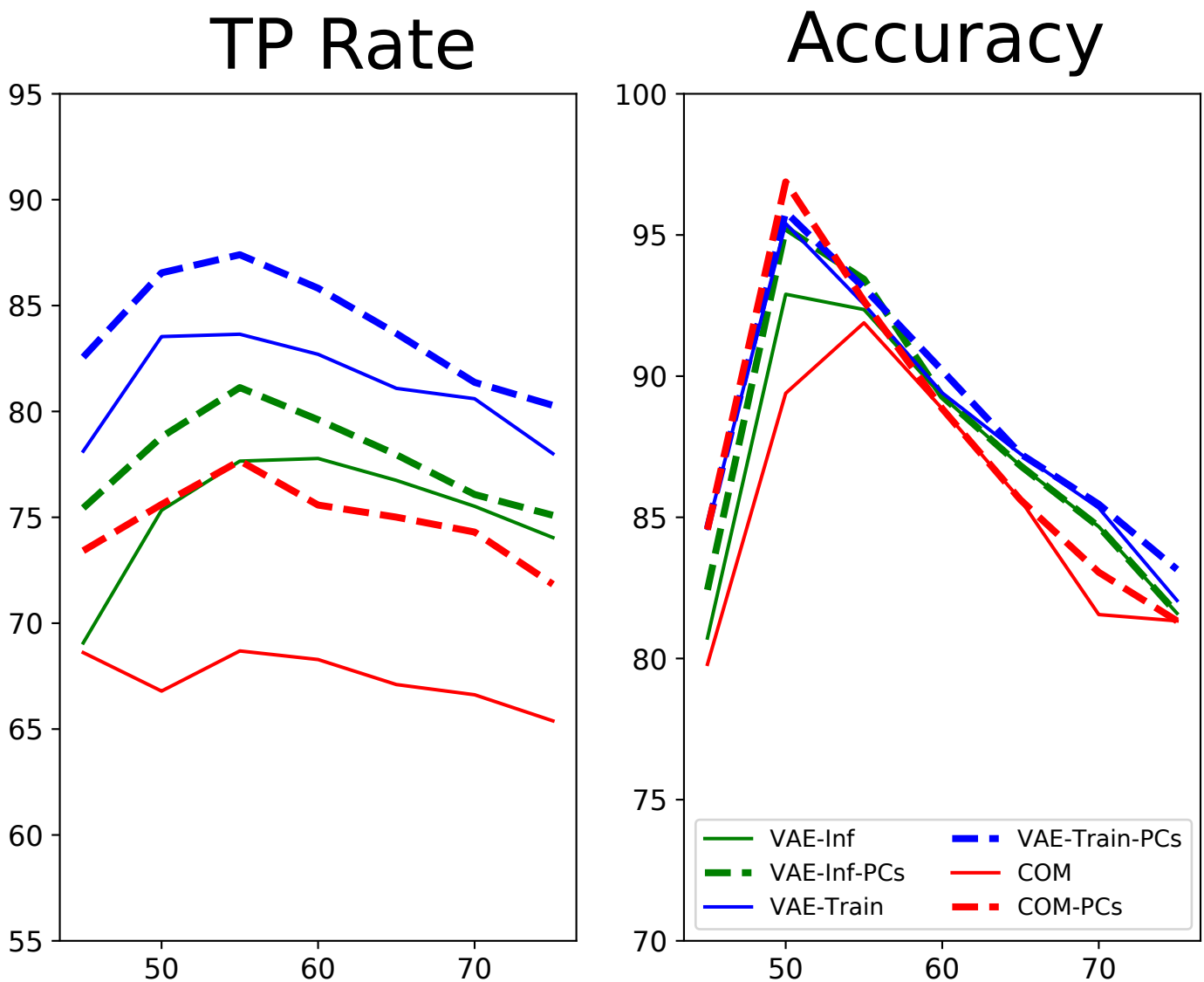
Below is the probabilistic program and inference code for the MCMC version of our method in Turing.

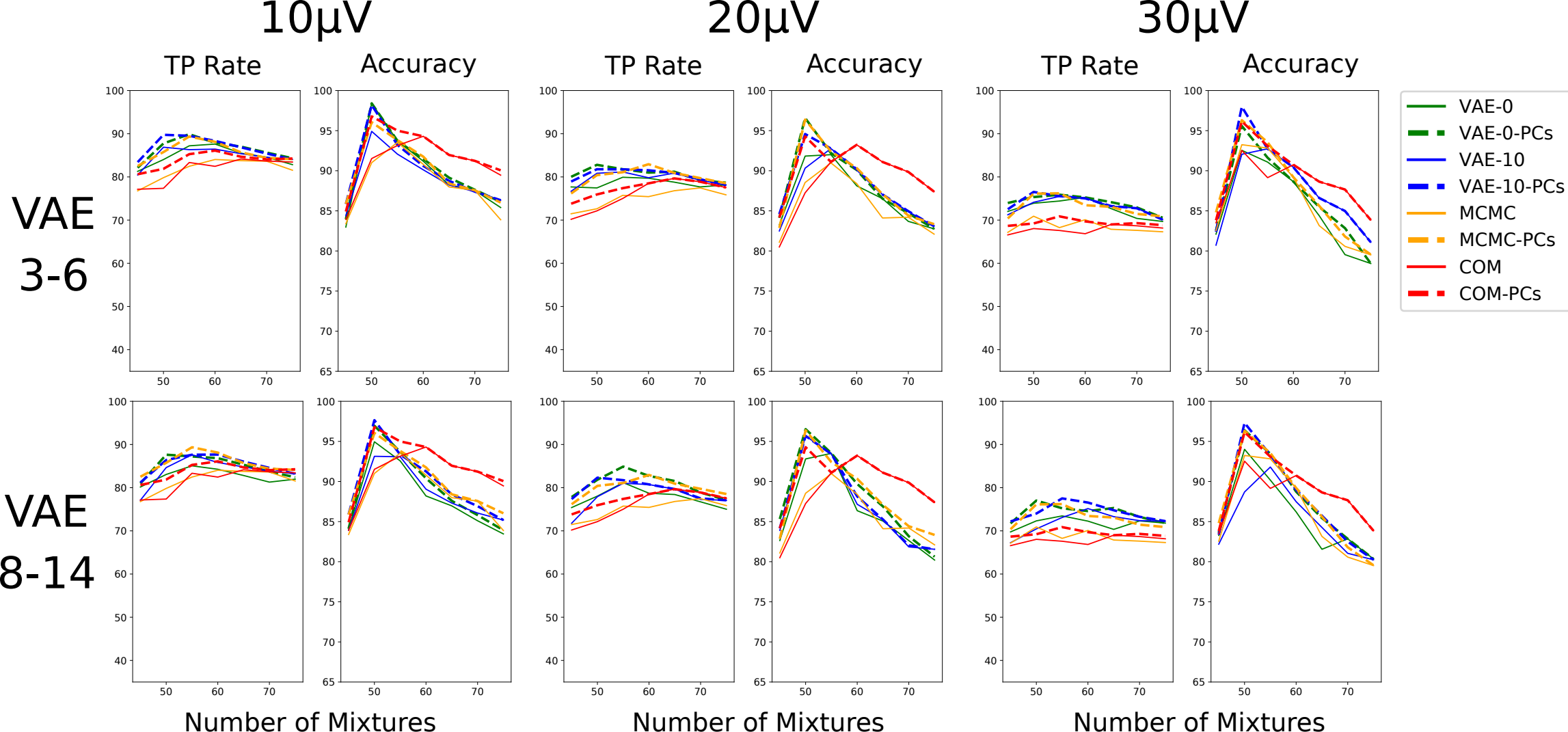
```
using Turing

# Define model
@model BayesianExpSpike(x_0, y_0, z_0, p_mean, p_std, locs, amps) = begin
    S_a = 1
    n_x ~ Normal(x_0, 80)
    n_y ~ Normal(y_0, 80)
    n_z ~ Normal(z_0, 80)

    a ~ Normal(p_mean[1], p_std[1])
    b = 0.035

    r = sqrt.(sum((abs.(locs .- [n_x; n_y; n_z;])).^2; dims=1))
    amps ~ MvNormal(vec(-a .* exp.(-b .* r)), S_a^0.5)
```

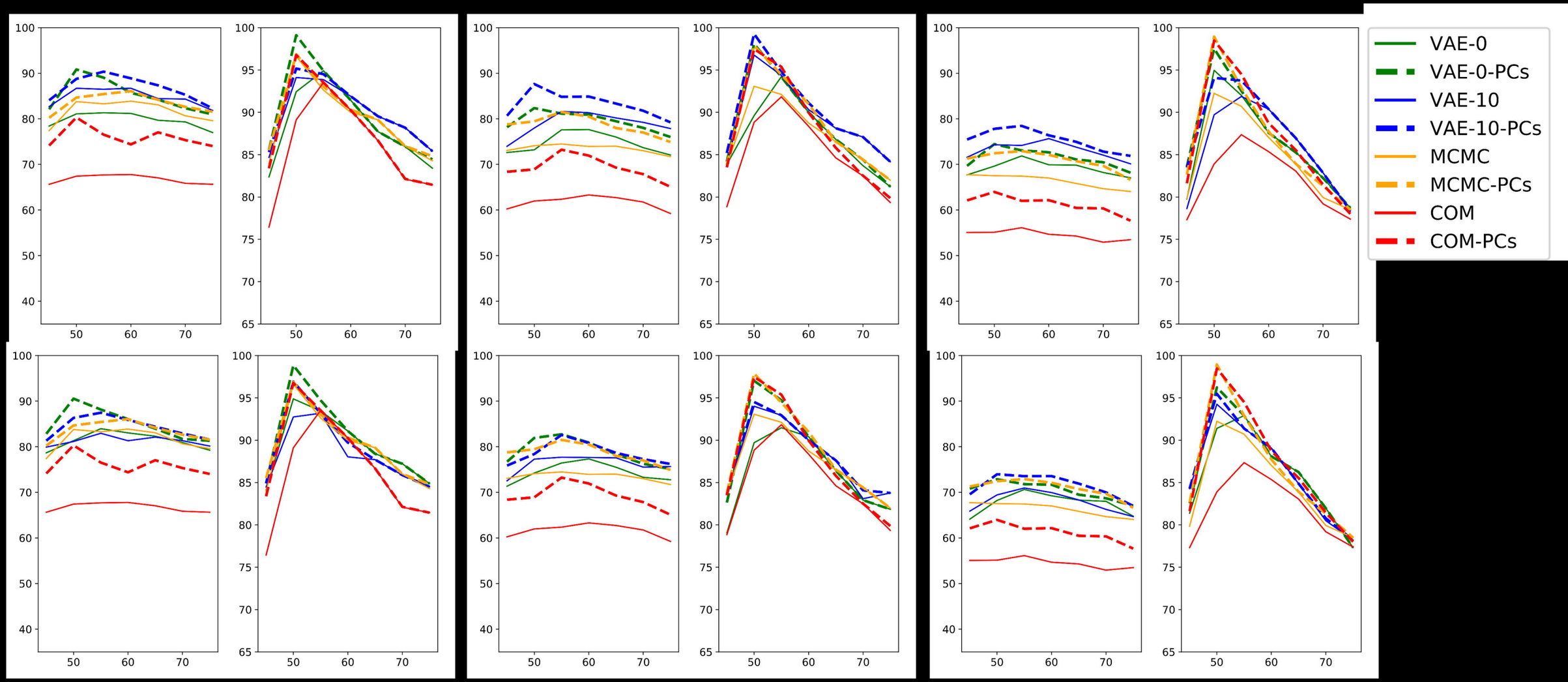
```
end

# Load data
real_channel_locs = ...
real_amps = ...
min_amp = ...
p_m = [abs(min_amp)*2]
p_s = [50]

# Feed data into model
model_func = BayesianExpSpike(0, 0, 0, p_m, p_s, real_channel_locs, real_amps)

# Sampling
chn = sample(model_func, HMC(10000, 0.01, 10));
```

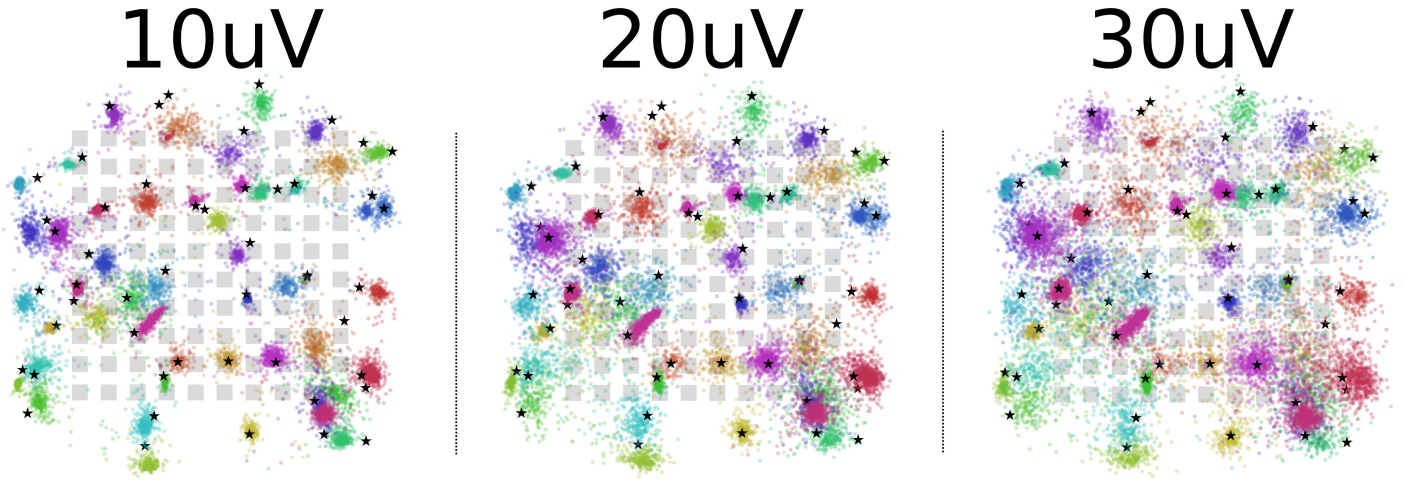
10μV

VAE
9-25

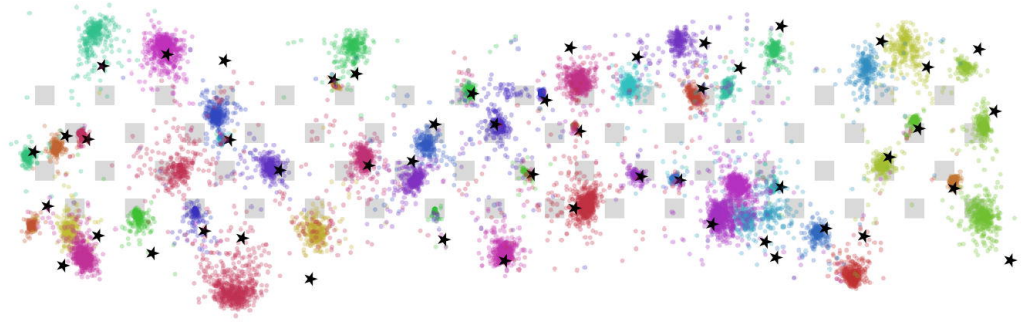

TP Rate
Accuracy
95
90
85
80
75
70
65
60
55
100
95
90
85
80
75
70
50
60
70
VAE-Inf
VAE-Inf-PCs
VAE-Train
VAE-Train-PCs
COM
COM-PCs


10μV
20μV
30μV
TP Rate
Accuracy
VAE 3-6
VAE 8-14
Number of Mixtures
VAE-0
VAE-0-PCs
VAE-10
VAE-10-PCs
MCMC
MCMC-PCs
COM
COM-PCs

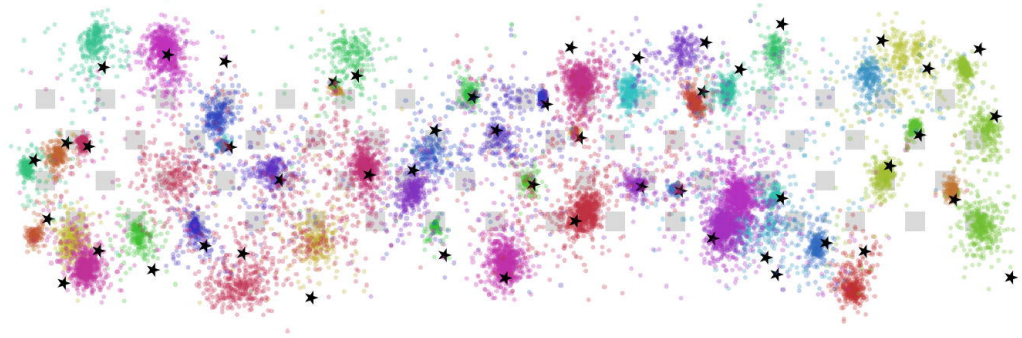

VAE-0
VAE-0-PCs
VAE-10
VAE-10-PCs
MCMC
MCMC-PCs
COM
COM-PCs


10uV
20uV
30uV

10uV

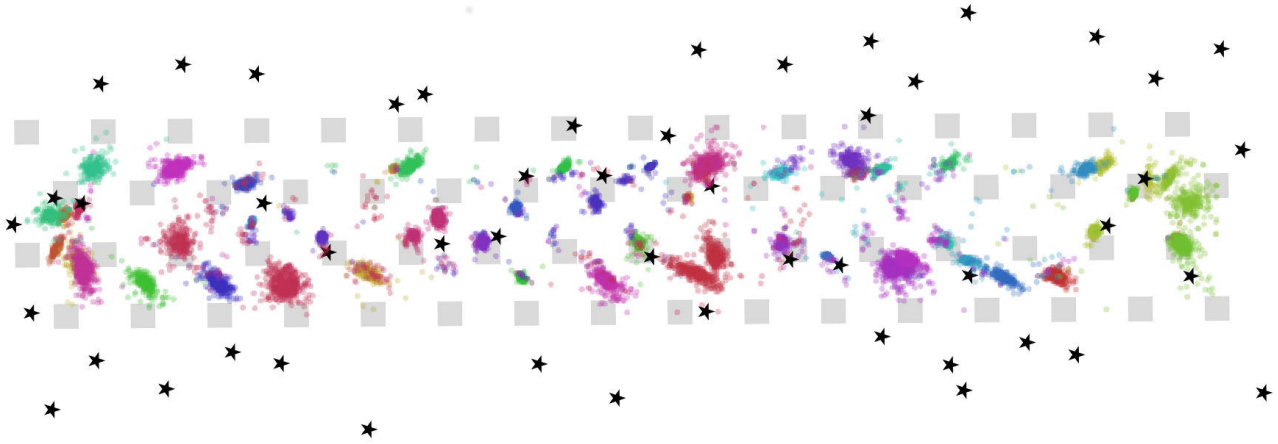


20uV

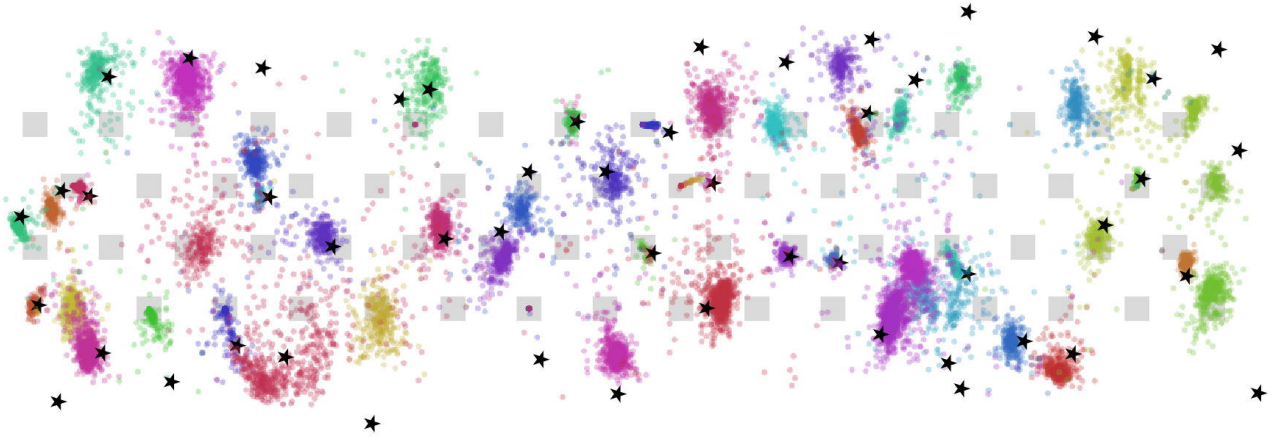


30uV

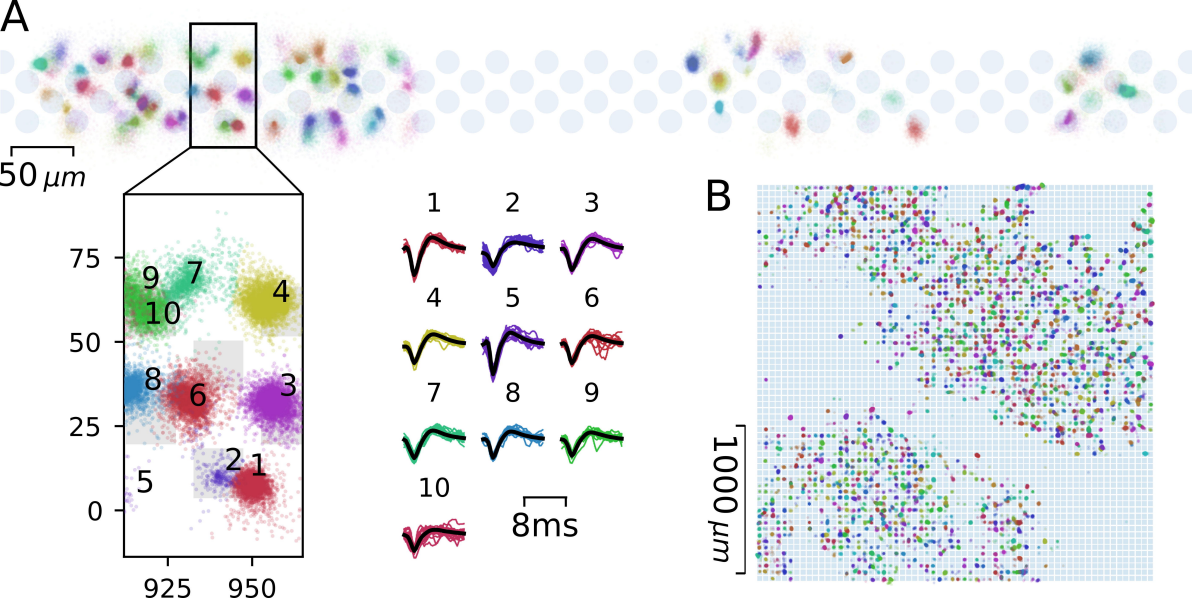
COM

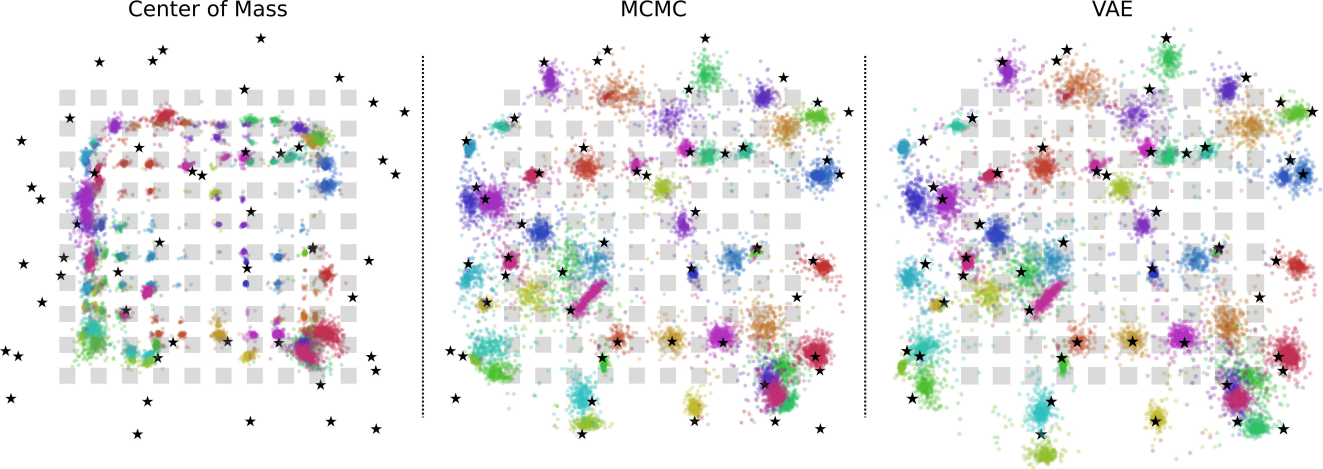


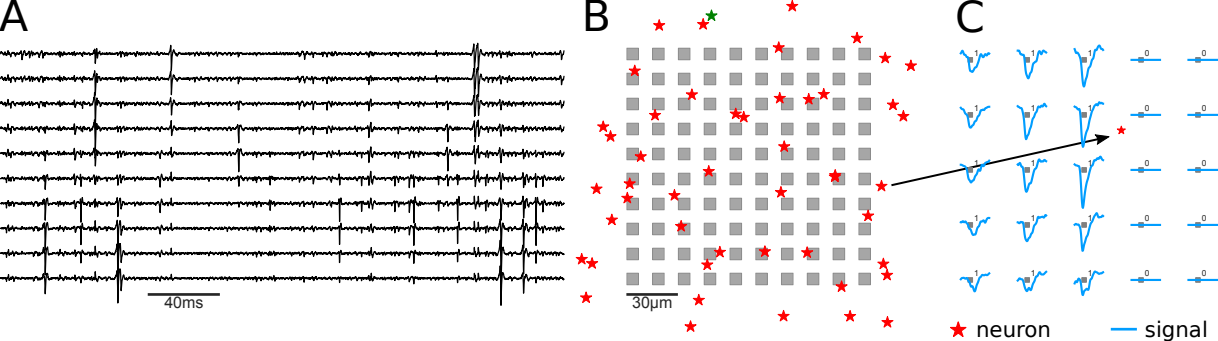
MCMC



VAE


A
50 μm
75
50
25
0
925
950
9
7
10
4
8
6
3
5
2
1
1
2
3
4
5
6
7
8
9
10
8ms
B
1000 μm


Center of Mass
MCMC
VAE


A
40ms
B
30µm
C
neuron
signal