A Bayesian Account for Estimating the Number of Neurons during Spike Sorting

Kees van Rooijen*, Bernhard Englitz*,

Department of Neurophysiology, Donders Center of Neuroscience, Radboud University, Nijmegen, The Netherlands

* vanrooijen@neurophysiology.nl, b.englitz@donders.ru.nl

Abstract

Extracellular recordings have long been an invaluable tool for understanding neural population activity. Spike sorting is the process of unmixing the contributing sources in a recording to obtain the spiking activity of individual neurons. Identifying the correct number of neurons is an error-prone process involving a considerable amount of intrinsic uncertainty. However, most spike sorting algorithms do not account for this uncertainty, but instead use a single point estimate. Using a fully probabilistic approach, we demonstrate that the point estimate leads to systematic misestimation of the number of neurons.

We estimate the number of neurons present in the data by sampling from the actual posterior distribution using reversible jump Markov chain Monte Carlo, in the context of realistic ground truth data. The expected value of the probabilistic estimate is then compared to the widely used maximum a posteriori (MAP) estimate of the number of neurons.

We find that even in the absence of incorrect modelling assumptions, using a point estimate leads to a systematic underestimation of the number of present neurons. This effect is visible for a wide range of values for the recording time and the noise available in the recording. More specifically, we find that decreasing noise leads to a decrease in this bias only for high sorting accuracy. If the sorting accuracy is low, this effect is reversed. Furthermore, we find that the size of the bias can initially be decreased by increasing the recording time, but for longer recordings this effect comes to a halt.

Misestimating the number of neurons contributes to errors in dividing spikes into clusters, and thus impacts the clarity of the results, e.g. by fusing different neurons, or splitting single neurons. As a consequence, correlations and other estimated properties would be affected. The present results provide an analytical guide to correct for this error.

Introduction

Nervous systems encode and process information on the basis of large networks of neurons. To gain an understanding of their structure and underlying mechanisms, it is essential to study their activity under realistic conditions in the context of stimulation and behavior. In particular the coordinated activity of multiple neurons is revealing about the patterns in and connections of the network. A classical yet versatile technique to gather this information are microelectrodes, placed inside the neural tissue in close proximity to the neurons. The recorded voltage activity is a mixture of many neural and nonneural sources, dominated by the signals from nearby cells.

The process of unmixing this composite signal and obtaining spiking activity from individual neurons is referred to as *spike sorting*. It is performed fully or semi-automatically and generally consists of three steps: (1) detecting putative spikes when the voltage trace or its derivative crosses a threshold, (2) extracting waveform features either directly such as amplitude or shape or after dimensionality reduction, (3) clustering spikes into separate units that represent individual neurons (see [1,2] for more details).

Determining the appropriate number of clusters can generally be done with a lot of certainty for close-by neurons, but the uncertainty gradually increases as neurons are further away from the electrode, or when neuron waveforms are relatively similar. Two neurons with similar waveform features may be mistaken for a single neuron, or spikes coming from a single neuron are separated into two different groups due to noise or slight shifts in waveform shape. Errors of this kind are severe and will directly influence the results of further analysis on the basis of the processed data [3,4]. Further, analysis of simulated data has shown that for more than 4 neurons actually present, commonly used spike sorting algorithms systematically underestimated the number of neurons [5].

A number of explanations for this systematic underestimation have been proposed, such as tissue damage caused by the electrode, low firing rates in nearby neurons and limitations of spike sorting algorithms. Incorrect modelling assumptions, such as Gaussian cluster shape, could lead to further mistakes [6]. We propose, however, that a biased estimation exists even in the absence of biological complications or incorrect modelling assumptions. In most spike sorting procedures the maximum a posteriori (MAP) estimate is used, i.e. given a set of spikes, only the most likely number of neurons is considered. We propose that this method, as opposed to modelling the existing uncertainty and considering multiple options for the number of neurons, by itself introduces a bias in any results obtained.

To arrive at these conclusions, we simulate data from a simplified model, with properties similar to real data. Unlike for real recordings, the generative model underlying the data is known in this case, enabling us to perform perfect, model-based probabilistic inference. Furthermore, the ground truth regarding the number of neurons is known, which allows us to compare the results obtained by using the point estimate with reality. We find that in this simplified but realistic scenario, the widely used point estimator systematically underestimates the number of present neurons.

Methods

In order to precisely understand the factors influencing the estimation of the number of neurons, we perform a fully Bayesian analysis to obtain the posterior distribution over the number of neurons, N_{cells}. This requires the specification of a generating model, including physiologically derived priors on $N_{\text{cells}}$and spike amplitudes, and a numerical determination of the posterior of N_{cells}. The latter is at the heart of the present study's contribution, as it requires a reversible jump Markov chain Monte Carlo (RJMCMC) implementation which can handle changes in the number of dimensions (proportional to N_{cells}).

Overview of the computational approach

Generally, spike sorting consists of four main steps. First, the contribution of the spikes and the local field potential to the voltage trace are separated by applying a band pass filter. Then, a detection threshold is set to identify the occurrences of the spikes. After the spikes have been identified, waveform characteristics are extracted for each spike

separately, and finally the spikes are clustered together based on the similarity of their characteristics. [2, 7]

In our investigation the focus is on the determination of the number of clusters in this final step. To isolate the error introduced in this step, we directly work with the extracted features, generated on the basis of physiologically informed priors. This frees us from potentially interfering contributions arising in spike detection and feature extraction.

Our procedure for simulating spike data therefore consists of three steps. First, generate the number of neurons that contribute to the signal. Second, for each neuron choose the firing rate and the distance from the electrode. Third, generate a number of spikes for each neuron. The number of generated spikes depends on the firing rate and the recording time. In the beginning, each spike is described by just one feature, the maximal amplitude, which depends on the neurons distance from the electrode and whose units are scaled to the overall noise level.

For reestimating the number of neurons, we discard the knowledge about the parameters that were used in generating the data. We estimate the number of neurons that were used for generating the data, using the extracted features of each spike and the prior knowledge. Unlike in the case of real recordings, the exact prior probabilities are known here and can be taken into account. Using this knowledge of the generative model, prior probabilities and the spike amplitudes, we can approximate the exact posterior probability distribution for the number of neurons.

Most spike sorting methods then determine the number of neurons as a point estimation, for example via a maximum-a-posteriori (MAP) estimate, i.e. the number of neuron with the largest posterior probability. As a control, we compute the probability-weighted average of the number of neurons, i.e. the expected value based on the posterior distribution. The procedure of generating the data and estimating the probabilities is repeated many times, so that the general behavior of both methods can be studied.

The choice of probability distributions and parameters used for the generations, as well as the specific sampling procedure are detailed below.

Generative model

The common approach to simulation of extracellular recordings is to superimpose waveforms of different neurons in time to obtain a realistic signal [8]. As indicated above, we instead directly generate the features for a given recording on the basis of physiological constraints.

Number of neurons

To determine the prior probability distribution of the number of active neurons in a recording, we need to consider the expected total number of neurons around the electrode and the fraction that is active. To find the total number, we use a tetrode range of 50 μm, the maximum range at which single units can be detected [9] and a density of 300.000 neurons/mm^3 [10]. Less is known about the fraction of neurons that is active during a recording, which is highly dependent on the region and the stimulus [10, 11]. For our simulations we chose a fraction of 0.03 so that the resulting distribution of N_{cells}matches common numbers found in extracellular recordings.

The distribution on the number of active neurons can be well approximated by a binomial distribution with $n = 24$ and $p \approx 0.2$ (see Figure 1A).

$$N_{\text{cells}} \sim \text{Binomial}(n, p). \tag{1}$$

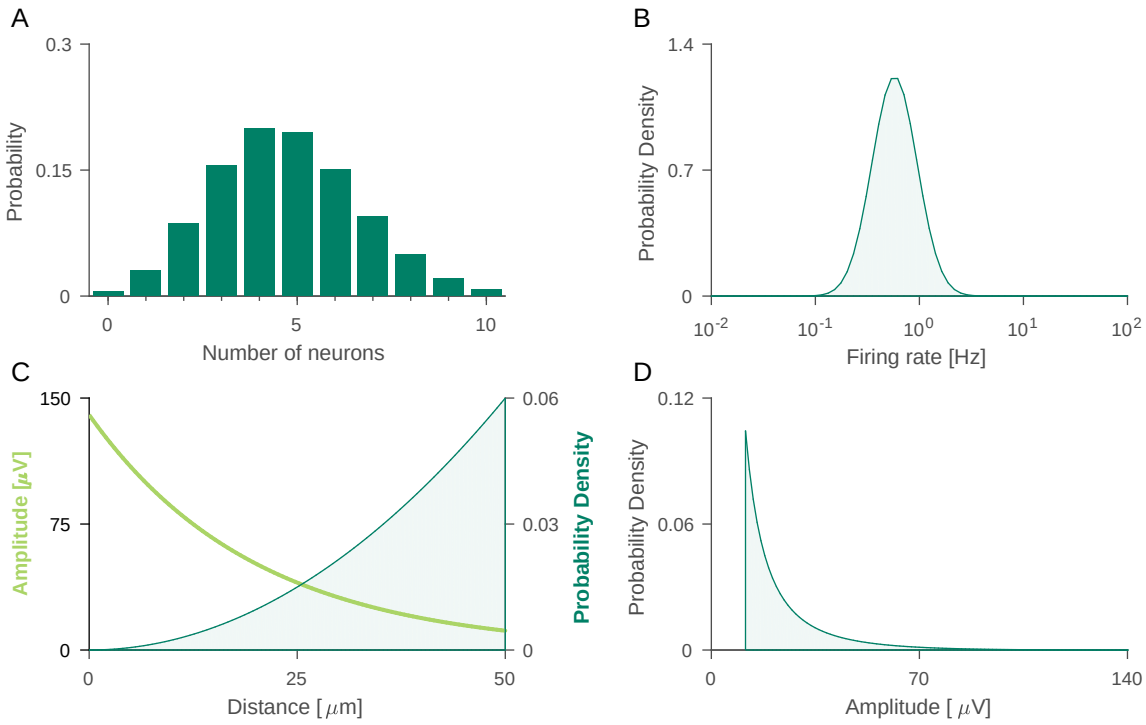

Fig. 1. The generative model is based on distributions resembling actual neuron characteristics. (A) The number of neurons in a range of 50 μm from the electrode, that are active during the recording, is well approximated by a binomial distribution with $n = 24$ and $p \approx 0.2$. **(B)** The firing rate of an active neuron in our simulations is given by a log-normal distribution, with $\mu = -0.3$ and $\sigma = 0.5$. **(C)** The relation between the distance d of a neuron from the electrode and the amplitude A of that neuron's spikes at the electrode *(left, light green)* is well described by the exponential curve $A = 140 * \exp(-d/20)$.The probability density of an active neuron being at distance d from the electrode is proportional to d^2 *(right, dark green)* . **(D)** Combining the findings of C, we find that the probability density of a neuron spiking with amplitude A is proportional to $A^{-1}\ \ln^2(A/140)$.

Amplitude at the electrode

When an electrode is placed in the extracellular space between neurons, the spiking activity of all neurons that are nearby will contribute to the voltage trace measured at the electrode. The contribution of a neuron depends on its distance from the electrode. Neurons closer than 50 μm will show up with large amplitudes and distinct waveforms, and can usually be well separated. Neurons closer than 140 μm will still be detectable, but their spikes not separable due to their relatively low contribution compared to the noise level. Spikes from neurons further away will no longer be detectable and only contribute to the noise [2].

Under the assumption that all neurons generate spikes with equal amplitudes, the average spike amplitude A of a neuron at the electrode depends only on the distance d of the neuron from the electrode. Following Coulomb's law, this relation is commonly modelled as $A \propto 1/d$ [8]. However, the relation has been experimentally approximated using a more sophisticated model of the neuron and was found to be well fit by an exponential curve

$$A = A_0 \exp(-d/20) \tag{2}$$

where A_0 corresponds to the spike amplitude of a cell at zero distance [10, 12].

Using this relation, we can transform the prior on the distance from the electrode and obtain the prior on the measured amplitude. Assuming that active neurons are uniformly distributed around the electrode, the probability density function of a neuron being at a certain distance d equals

$$p(d) \propto d^2 \tag{3}$$

for $0 \leq d \leq 50$ (see Figure 1C for these distributions when $A_0 = 140\mu V$). Combining these we obtain

$$p(A) \propto A^{-1} \ln(A/A_0)^2 \tag{4}$$

with possible amplitudes ranging from $A = A_0 \exp(-50/20)$ to $A = A_0$ (see Figure 1D).

Firing rate

The firing rate f of an active neuron can be best described using a log-normal distribution [13].

$$\log(f) \sim \mathcal{N}(\mu_f, \sigma_f) \tag{5}$$

The exact parameters μ_f and σ_f describing this distribution are highly dependent on the region and behavioural state, but the shape is consistent. For our simulations we used values $\mu_f = -0.3$ and $\sigma_f = 0.5$ (see Figure 1B).

Spike generation

The number of spikes k generated by each neuron is determined by its firing rate and the total recording time τ_{rec},

$$k \sim \text{Poisson}(f\ \tau_{\text{rec}}) \tag{6}$$

Then, k spikes are drawn from a normal distribution around the neuron's mean amplitude, using the overall noise variance of the recording.

$$x \sim \mathcal{N}(A, \sigma_{\text{noise}}) \tag{7}$$

Lastly, the final data are obtained by concatenating the spikes from all simulated neurons. The resulting list consists of all the amplitudes of all spikes present in the recording. No additional information about the number of neurons present in the recording and their characteristics is available for the sorting process.

Sampling from the posterior distribution

We approximate the posterior probability distribution on the number of neurons N_{cells} available in the recording, using only the data X and the prior knowledge of the model, using Bayes' theorem.

$$P(N_{\text{cells}}|X) \propto P(X|N_{\text{cells}})\ P(N_{\text{cells}}) \tag{8}$$

To approximate the posterior we sample the number of neurons and the characteristics of each neuron. The sampling space is complicated in the sense that the dimension depends on the number of clusters. We implemented a dedicated reversible jump Markov Chain Monte Carlo sampler (RJMCMC) [14, 15], which allows for jumps between states with different dimensionality. Instead of sampling directly the neurons present in the recording, we use a high, fixed amount of neurons, of which a large part is inactive and does not produce spikes. To obtain the actual number of neurons present in the recording, we count only the active neurons. This simplifies the sampling space

by fixing the dimension, but greatly increases the number of parameters that need to be sampled.

To prevent a large increase in computation time, we optimized the code so that it keeps track of all probabilities that were computed before, so the parameters of each neuron can be updated without the need to recompute probabilities for every other neuron. The sampling of the parameters of an inactive neuron now effectively becomes a *birth step* in a traditional RJMCMC. We sample for a total of 1000 iterations, of which 400 are used as a burn-in period. We find that these settings produce accurate and reproducible results.

The computational procedure consists of repeatedly generating data and sampling posterior. Because each run is completely independent, it was straightforward to parallelize the code, further reducing the running time of the program. The final code will be made available alongside the publication.

After sampling, we count the active neurons in each iteration to obtain the probabilities on the number of neurons. The probabilistic (PROB) method computes the expected number of neurons

$$\hat{N}_{\text{cells,PROB}}(X) = \sum_n n * P(n|X). \tag{9}$$

The maximum-a-posteriori (MAP) method can simply use the mode of the posterior distribution

$$\hat{N}_{\text{cells,MAP}}(X) = \arg\max_n P(n|X). \tag{10}$$

Statistical procedure

To compare both methods of estimation, we used the Wilcoxon Signed Ranks test. All errorhulls depict standard deviations, since standard errors of the mean (SEM) are less interesting in simulations, where the number of simulations would eventually make the SEM go to 0.

Results

To understand how the MAP estimate introduces a bias, it is informative to observe how the probabilities behave as a function of the data. First, consider a few feature samples produces by different runs of the generative model (Figure 2A). They differ in the overall signal size as well as the distributional shape, based on the number and relative location of the contributing neurons. The number of spikes and the noise present in the signal limit the certainty with which the distributions are known. We estimate the posteriors using the RJMCMC algorithm (Figure 2B), which peak at the number of peaks that each feature sample has, but in addition illustrate that the probability for a different number of neurons is not vanishing. This means that the samples could as well have been generated by a different number of neurons than suggested by the peak of the posterior distribution (i.e. the MAP estimate).

Concretely, if two clusters with different amplitudes are separable, it is very unlikely that they have been generated by the same neuron, and we see that the probability of having one neuron is very close to zero (Figure 2A/B; top two rows). If one cluster of spikes with similar amplitudes exists, the solution with one neuron is more likely, but the option of two neurons with similar amplitudes will not be completely ruled out: the probability of having two neurons is small but clearly non-zero (Figure 2A/B; bottom row).

Generally, it is important to understand the trade-off in the computation of the posterior probability: more neurons always have the potential to explain the data

better, but to do so they require more parameters with a suitable value. This increased model complexity that having an extra neuron brings is somewhat constant and does not depend on the data variance, which ensures that generally simpler solutions with fewer neurons are preferred. However, as variance in the data increases or data is even clearly separable into two clusters, the gain in explanatory value of having two neurons outweighs the cost, and the solution with more neurons is preferred.

To quantify this effect over the full range of numbers of clusters, we centred the posterior probability distributions on their modes. In this way we can observe how the probabilities behave on both sides of the mode, for higher and lower numbers of neurons (Figure 2C). Probabilities on the right clearly show higher values than their counterparts on the right. From this we can understand that the MAP method will induce a bias: the expected difference between the actual number and the mode is larger than zero.

Next, consider evaluating the full probability distribution for estimating N_{cells}as a function of the actual number of cells (PROB, Figure 2D). When few neurons are used for data generation, the sampler will rarely underestimate this number. The probability of having more neurons is always non-zero, leading to a general overestimation of the number of neurons. When more neurons are present, the spikes from all neurons saturate the amplitude space. A solution with more neurons only gives a limited improvement. Less clusters are therefore preferred, and we observe a general underestimation. The shape of the curve can further be understood using Bayes' rule (Eq. 8). The expected number of neurons summarizes the posterior distribution. The posterior is obtained by combining the data with the prior knowledge. In figure 2D we show both the number of neurons used to generate the data (on the diagonal), and the expected number of neurons based on prior knowledge alone (as the dark green striped line). The expected number of neurons based on the full posterior can be found in between these lines, combining their results.

Finally, we can observe the underestimation of the MAP method compared to the PROB method. The underestimation is consistent over the full range of numbers of neurons (Figure 2E, $p < 0.0001$ for all numbers of neurons). This implies that the MAP method actually seems to be more accurate for lower numbers of neurons, where the full probabilistic estimations overestimates the number of neurons. Here it is essential to understand that the probabilities as found by the PROB method are the actual posterior probabilities. In the absence of ground truth, there is no estimate that is more accurate than the one provided by the PROB method. Although it is counter-intuitive at first, this means that the underestimation of the MAP method compared to the PROB method does not lead to an improvement for low numbers of neurons, because in the absence of ground truth data, the cases in which the number of neurons are low cannot be separated from cases with more neurons.

To arrive at the total expected underestimation in a single recording, we can combine the relative occurrence of each number of neurons with the expected underestimation for that number. The separation that we used so far, based on the number of neurons in the recording that we used so far, was used for illustrative purposes only. In real recordings the lack of ground truth precludes this comparison. For the exploration of the effect of different parameters, we consider all simulations together.

Noise and recording time

With the insight of how the bias comes to exist, we can explore the effect of the different parameters on the size of the bias. The size of the bias is obtained by combining the underestimation for each number of neurons with the prior probability for that number, yielding the total expected underestimation for one simulated recording.

The magnitude of the noise found in a recording is in principle a property of the brain and not up to the experimenter. However, the noise used in our simulation has a

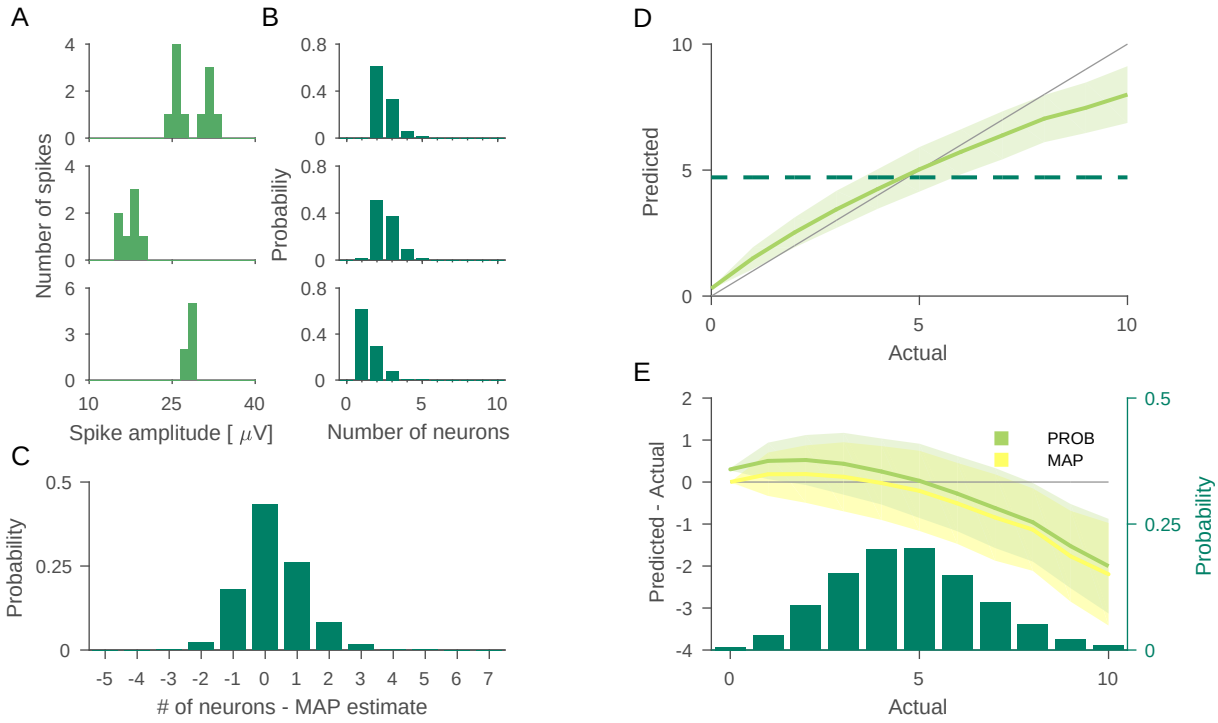

Fig. 2. The MAP method introduces a bias due to the asymmetry in the posterior distribution of $N_{\mathbf{cells}}$. Data presented result from simulations where $\sigma_{noise} = 1.0\mu V$ and $\tau_{rec} = 5.0s$. **(A)** Three selected examples of histograms of data generated by our generative model, and **(B)** the corresponding posterior probabilities as found by our sampler. **(C)** Posterior distributions as in (B) are centred on their mode, showing higher probabilities to the right of the mode than their counterparts on the left. **(D)** The number of neurons used in data generation versus the expected number of neurons as computed by the sampler (PROB). The dark green striped line represents the prior expectation. **(E)** *(left)* Difference between the number of neurons used in data generation and the expected number of neurons by either the MAP method or the PROB method. *(right)* The relative occurrence of each number of neurons in the repetitions of data generation.

direct relation with the amount of space a spike cluster takes in the feature space. Therefore, using more than one informative spike feature or using better recording techniques such as tetrodes could be modelled by decreasing the noise. In figure 3A we see that a lower amount of noise leads to more accurate results of both methods.

For each value for σ_{noise} in (A) and (B) we ran 5000 simulations, for each value of τ_{rec} in (C) and (D) we ran $\min(5000, 50000/\tau_{\text{rec}})$ simulations.

However, the relation between the noise and the bias induced by the MAP method is non-monotonic. For a higher noise, decreasing the noise actually increases the size of the bias. Only for very low noise values, when the accuracy reaches high levels, the bias decreases as noise values decrease (Figure 3B).

For recording time we see a different relationship: as the recording time and thus the number of spikes increases, the accuracy increases and the bias decreases (Figure 3C/D). However, although having a moderately high recording time is essential to obtain accurate results and low bias, increasing the recording time beyond that does not lead to a substantial improvement in accuracy or decrease in bias.

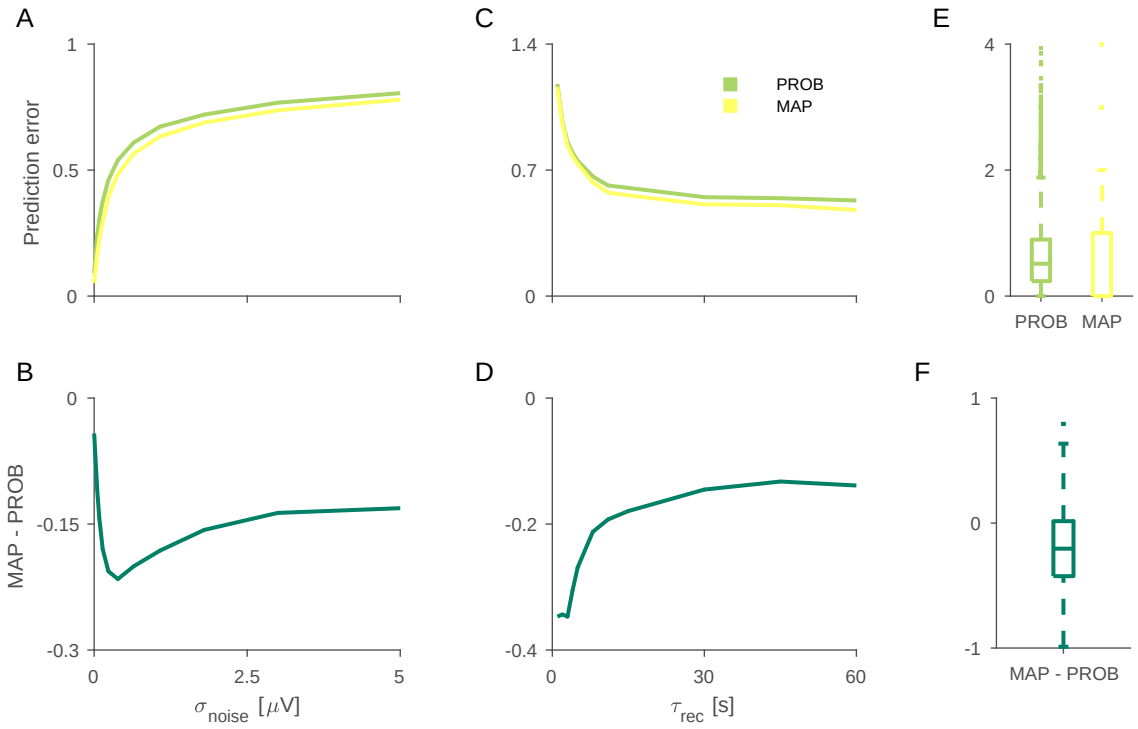

Fig. 3. The size of the bias depends non-monotonically on the recording noise and reduces with longer recording durations. (A) The sorting error decreases when signal noise decreases. The error is the absolute difference between the estimated number of neurons and the actual number used in data generation. **(B)** The bias decreases when noise gets very low, but for higher noise values this relation is reversed. For (A) and (B) we used $T_{\text{rec}} = 10\ s$. Note, that the prediction error and bias are aggregated over the range of actual neuron numbers. For short recording time, the sorting error **(C)** and bias **(D)** decrease as recording time increases. For even longer recording times, sorting error and bias remain stable. For (C) and (D) we used $\sigma_{\text{noise}} = 0.8\ \mu V$. **(E, F)** An example of the distributions of values. For these simulations we used $\sigma_{\text{noise}} = 0.8\ \mu V$ and $T_{\text{rec}} = 10\ s$.

Discussion

In the present study we investigated how estimating number of neurons during spike-sorting depends on utilized estimation technique. We find that the classical maximum-a-posteriori (MAP) technique systematically underestimates the number of neurons in comparison to the full information from the posterior, an effect that persists for different levels of noise and recording duration. On the one hand, this encourages the use of the full posterior for determining the number of neurons in extracellular recordings, but on the other hand also provides information about the intrinsic uncertainty in determining N_{cells}, which has substantial consequences for derived quantities, such as correlations between the neural responses [16].

Interpretation of the difference in estimation

The comparison of estimation techniques indicates that the MAP estimate systematically underestimates the number of clusters. When we use the ground truth to separate the results based on the number of neurons based in a recording, the

underestimation is present for the full range of neurons (Figure 2D). At first glance this appears to render the estimate more accurate for at least a part of the range of possible values of N_{cells}. However, for actual recordings the ground truth is not available, and a set of spikes can arise from multiple real configurations (in this case the number of cells and their respective spike sizes). This makes it impossible to know when it would be correct to use the MAP-estimate.

If we remove the separation based on the ground truth and thereby effectively average the underestimation over the range of numbers of neurons as described before (Figure 3), we find the expected bias that is introduced by using the MAP-estimate. Because the MAP-estimate is by definition the point estimate that uses the most likely option for every recording, we should only expect results to be worse for non-optimal point estimates.

Extendability to realistic scenarios

The currently used sorting algorithm uses only a single feature, the spike amplitude. In realistic sorting methods multiple features are used. Neurons are furthermore measured at multiple electrodes. Because the dimensionality of the feature space is increased, clusters of spikes with distinct features can be more easily separated. In our model this effect is captured in the recording noise parameter. Lower noise values give more space to distinct spike clusters in the (here: one-dimensional) feature space, yielding the same effect as increasing the dimensionality of the feature space itself.

Another difference with realistic spike sorting situations is the explicit modelling of the firing rates. Model-based sorting can provide accurate results on the number of neurons even in the presence of high recording noise, based on the number of spikes and the recording time. In realistic spike sorting situations, this knowledge may not be available, not be used, or could itself be the property that one is interested in. Not incorporating the available knowledge in the sorting process may lead to an additional bias in the results [4, 17]. Another difference between our simulations and reality is the availability of correct model priors. In common sorting algorithms, priors on waveform features are usually not explicitly modelled.

The explicit modelling of prior knowledge and available information makes that our model-based sorting process may show different behaviour and be more accurate than the sorting process in realistic scenarios. Therefore, the size of the bias that is introduced in spike sorting for the reasons proposed in this paper is difficult to estimate. It shows, however, that the bias exists, even in the absence of incorrect modelling assumptions or incomplete information. Furthermore, the observed underestimation patterns match the pattern that was experimentally observed [5].

Implications

A misestimation of the number of cells present in the recording will directly influence many derived properties in individual or between cells. The most obvious example is the firing rate: an underestimation of N_{cells} will lead to an overestimation of the average firing rate. Correlation is another property that is influenced by misclassification: Mistakenly assigning spikes of two neurons to one cluster or dividing spikes from one neuron into two clusters has distinct effects on the measured correlation (see [3] for a detailed overview). In the present case, the consequence of an underestimation of N_{cells} would be a greater likelihood of incorrectly fusing clusters: as a consequence the correlation of this cluster with other clusters could change.

Several attempts have been made to introduce model based sorting for spike sorting [18, 19]. Further, it has been proposed to perform the spike sorting process and the estimation of the properties of interest simultaneously, thereby making use of all

available knowledge in the sorting process [4, 17]. In the light of the current findings, we encourage the development of a general framework in which *a priori* assumptions about the generative model and the used parameters are made explicit, after which joint probabilistic inference can be performed on the latent variables and the properties of interest. This removes any kind of bias and provides a direct way to model uncertainty.

However, until such a unified framework is made available, other measures can be taken to prevent drawing conclusions based on spike sorting artifacts. One option would be to perform a bootstrapping procedure, generating new data samples from the acquired properties to estimate confidence intervals [20]. Although this technique will not necessarily yield statistically consistent results, it will provide a starting point for finding correct confidence intervals.

References

1. Einevoll GT, Franke F, Hagen E, Pouzat C, Harris KD. Towards reliable spike-train recordings from thousands of neurons with multielectrodes. Current opinion in neurobiology. 2012;22(1):11–17.
2. Rey HG, Pedreira C, Quiroga RQ. Past, present and future of spike sorting techniques. Brain research bulletin. 2015;119:106–117.
3. Cohen MR, Kohn A. Measuring and interpreting neuronal correlations. Nature neuroscience. 2011;14(7):811.
4. Ventura V. Traditional waveform based spike sorting yields biased rate code estimates. Proceedings of the National Academy of Sciences. 2009;106(17):6921–6926.
5. Pedreira C, Martinez J, Ison MJ, Quiroga RQ. How many neurons can we see with current spike sorting algorithms? Journal of neuroscience methods. 2012;211(1):58–65.
6. Shoham S, Fellows MR, Normann RA. Robust, automatic spike sorting using mixtures of multivariate t-distributions. Journal of neuroscience methods. 2003;127(2):111–122.
7. Lewicki MS. A review of methods for spike sorting: the detection and classification of neural action potentials. Network: Computation in Neural Systems. 1998;9(4):R53–R78.
8. Martinez J, Pedreira C, Ison MJ, Quiroga RQ. Realistic simulation of extracellular recordings. Journal of neuroscience methods. 2009;184(2):285–293.
9. Buzsáki G. Large-scale recording of neuronal ensembles. Nature neuroscience. 2004;7(5):446.
10. Henze DA, Borhegyi Z, Csicsvari J, Mamiya A, Harris KD, Buzsáki G. Intracellular features predicted by extracellular recordings in the hippocampus in vivo. Journal of neurophysiology. 2000;84(1):390–400.
11. Shoham S, O'Connor DH, Segev R. How silent is the brain: is there a "dark matter" problem in neuroscience? Journal of Comparative Physiology A. 2006;192(8):777–784.
12. Gold C, Henze DA, Koch C. Using extracellular action potential recordings to constrain compartmental models. Journal of computational neuroscience. 2007;23(1):39–58.

13. Buzsáki G, Mizuseki K. The log-dynamic brain: how skewed distributions affect network operations. Nature Reviews Neuroscience. 2014;15(4):264.

14. Green PJ. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. Biometrika. 1995;82(4):711–732.

15. Mazet V, Brie D. An alternative to the RJMCMC algorithm. In: 2006 IAR Annual Meeting, Nancy, France. IAR; 2006. p. CDROM.

16. Ecker AS, Berens P, Keliris GA, Bethge M, Logothetis NK, Tolias AS. Decorrelated neuronal firing in cortical microcircuits. science. 2010;327(5965):584–587.

17. Ventura V. Automatic spike sorting using tuning information. Neural Computation. 2009;21(9):2466–2501.

18. Wood F, Black MJ. A nonparametric Bayesian alternative to spike sorting. Journal of neuroscience methods. 2008;173(1):1–12.

19. Gasthaus J, Wood F, Gorur D, Teh YW. Dependent Dirichlet process spike sorting. In: Advances in neural information processing systems; 2009. p. 497–504.

20. Efron B. Bootstrap methods: another look at the jackknife. In: Breakthroughs in statistics. Springer; 1992. p. 569–593.