Relevance of Nonlinear Dimensionality Reduction for Efficient and Robust Spike Sorting

Daniel Suarez-Barrera[1], Lucas Bayones[1], Sergio Parra[1,2], Viktor Monroy[1], Sebastián Pujalte[1], Bernardo Andrade-Ortega[1], Héctor Díaz[1], Manuel Alvarez[1], Antonio Zainos[1], Alessio Franci[2,3*], Román Rossi-Pool[1,4*]

[1]Instituto de Fisiología Celular ─ Neurociencias, Universidad Nacional Autónoma de México, 04510 Mexico City, Mexico

[2]Montefiore Institute, University of Liège, Av de la Découverte 10, 4000 Liège, Belgique

[3]WEL Research Institute, avenue Pasteur, 6, 1300 Wavre, Belgique

[4]Centro de Ciencias de la Complejidad, Universidad Nacional Autónoma de México, Mexico City, Mexico.

*Correspondence: alessio.franci83@gmail.com (A.F.); romanr@ifc.unam.mx (R.R.-P.)

Abstract

Spike sorting is one of the cornerstones of extracellular electrophysiology. By leveraging advanced signal processing and data analysis techniques, spike sorting makes it possible to detect, isolate, and map single neuron spiking activity from both *in vivo* and *in vitro* extracellular electrophysiological recordings. A crucial step of any spike sorting pipeline is to reduce the dimensionality of the recorded spike waveform data. Reducing the dimensionality of the processed data is mandatory to apply the clustering algorithms responsible for detecting, isolating, and sorting the recorded putative neurons. In this paper we propose and illustrate on both synthetic and experimental data that employing the nonlinear dimensionality reduction technique Uniform Manifold Approximation and Projection can drastically improve the performance, efficiency, robustness, and scalability of spike sorting pipelines without increasing their computational cost. We show how replacing the linear or expert-defined dimensionality reduction methods commonly used in spike sorting pipelines by unsupervised, mathematically grounded, nonlinear methods increases the number of correctly sorted neurons, makes the identification of quieter, seldom spiking neurons more reliable, enables deeper and more precise explorations and analysis of the neural code, and paves new ways toward more efficient and end-to-end automatable spike sorting pipelines of large-scale extracellular neural recording as those produced by high-density multielectrode arrays.

INTRODUCTION

Accurately sorting individual neuronal spikes from large-scale recordings is key to understanding how neural activity encodes information in cognitive and sensory functions (1–3), tracing the flow of information across neural networks (4), and examining communication between population subspaces (5). High-density extracellular electrode array technologies, such as fully integrated silicon (6), are enabling the simultaneous recording of hundreds to thousands of neurons (7–9). Despite this progress, a vast majority of spike sorting methods rely heavily on manual parameter tuning to ensure accurate performance. The need for fast, automatic, and accurate spike sorting methods is now stronger than ever.

To ensure efficient clustering of spiking waveforms, a necessary step in any spike sorting pipeline is to reduce the dimensionality of the processed data. Traditional approaches often employ linear dimensionality reduction techniques, such as principal component analysis (PCA) (e.g., Herding (10), Klusta (11), Tridesclus, YASS (12), SpiKING

CIRCUS (13), SpikeDeep (14), Perceptron (15)), or Wavelet transform-based methods (e.g., WaveClus (16), WimSorting (17)), supplemented by ad-hoc nonlinear metrics like energy or peak-to-peak measures (16, 18). Existing approaches to spike sorting that include a nonlinear data dimensionality reduction step are usually based on deep-learning methods, like autoencoders and convolutional perceptrons (e.g., Autoencoders (19), 1dCNN (20), Perceptron (15)). Finally, powerful toolkits like KiloSort (21, 22) and algorithms tailored to large-scale electrode arrays (13) are constantly being developed to address the problem of achieving fast and accurate spike sorting from hundreds of spatially correlated extracellular recordings.

In all the aforementioned methods, the critical step of reducing the dimensionality of spiking data is either based on linear methods, ad-hoc nonlinear metrics, or deep-learning methods. For none of these dimensionality reduction methods there exist rigorous, mathematically grounded guarantees that fundamental geometric and topological properties of the original data, and most importantly the existence of clusters associated with spikes from different neurons) are preserved in the reduced data.

The Uniform Manifold Approximation and Projection (UMAP) algorithm (23, 24) has recently been introduced to perform data dimensionality reduction with rigorous guarantees that the reduced data preserves key geometric and topological properties of the original data. By preserving and revealing in low-dimensional projections geometric and topological properties of high-dimensional datasets, UMAP has already proved successful in uncovering the low-dimensional geometry of neuronal population responses (25) and in efficiently identifying and classifying cell types (26–29). Besides, UMAP needs little parameter tuning, is insensitive to noise and outliers, and possesses good scalability when faced with large datasets. These reasons is why UMAP fits as a good candidate for the application of data dimensionality reduction in spike sorting pipelines.

Both features make UMAP particularly suited for spike sorting. The nonlinear structures identified by UMAP serve as self-discovered geometric (nonlinear) criteria that distinguish spike waveforms with high accuracy, requiring no user intervention. Moreover, UMAP's robustness to data point density ensures that sparsely represented spike waveforms—such as those from neurons with low firing rates— can efficiently be identified and separated from densely represented waveforms originating from neurons with high firing rates. In contrast, methods that rely on density or variance metrics (13, 16) necessarily struggle to isolate low-firing rate neurons. Consequently, many

conventional sorting methods tend to overlook quieter neurons, leading to a significant loss of potentially valuable spiking information (30–32).

All the mentioned features of the UMAP algorithm are particularly relevant when dealing with hundreds or thousands of signals measured in a single experiment. As the number of electrodes in an array increases, methods that require parameter hand-tuning for guaranteed performance become increasingly less usable. For example, using a 64-electrode multiarray throughout a week of recordings generates more than 300 different datasets, each consisting of a variable number of identifiable neurons. With dense multi-electrode arrays or multi-site recording, it is possible to record from hundreds or even thousands of electrodes simultaneously (7–9). Therefore, an unsupervised and efficient sorting method is crucial for isolating single-neuron responses, including silent ones, from these massive databases.

In this study, we introduce a novel spike-sorting pipeline based on UMAP (23). We apply this pipeline to identify single-neuron responses in several datasets, including synthetic data (16), electrophysiological data with ground truth (33), and in-vivo recordings during cognitive tasks (34). Our method shows significant improvements in accuracy compared to existing spike sorting methods such as (35). The proposed spike sorting method stands out in particular for its drastically improved robustness to even large heterogeneities in the recorded neuron firing rates. UMAP-based spike sorting effectively isolates and identifies low-firing rate and even almost silent neurons. Conversely, spike sorting pipelines based on linear dimensionality reduction methods or expert-defined metrics tend to either dilute the spike waveforms from low-firing rate neurons into the clusters associated with high-firing frequency ones or simply to ignore them. Furthermore, the unsupervised nonlinear nature of the UMAP algorithm allows it to isolate neurons with spike waveforms that look indistinguishable to classical methods, based on PCA or Wavelet transform, and that can be challenging to distinguish even through expert-defined metrics. To perform unsupervised sorting of recorded cells from UMAP projections, we employ hierarchical density-based spatial clustering application with noise (HDBSCAN) (36, 37). HDBSCAN is a clustering algorithm that does not require to pre-specify the sought number of clusters and can automatically detect data points that cannot be clustered by labelling them as “noise”.

We show the superior performance of the UMAP-based spike sorting pipeline using minimal parameter tuning in a variety of settings. The proposed pipeline builds upon but also drastically improves existing spike sorting techniques and technologies (6, 13, 21,

35). Crucially, we highlight the ability of UMAP-based spike sorting to identify more neurons, particularly low-firing rate ones, as compared to previous methods, which in turn enable more accurate analysis of the information encoded in spikes both at the single neuron and population level. To summarize, we present a novel unsupervised, data efficient, and high-performance pipeline, grounded on nonlinear dimensionality reduction methods, to sort individual neuron spikes from both standard and high-density recordings.

RESULTS

UMAP-based spike sorting vs. traditional feature-based pipelines

Figure 1 provides an overview of the proposed UMAP-based spike sorting pipeline, highlighting both its commonalities with existing methods and the unique steps that distinguish it. The four initial steps—high-pass filtering (Fig. 1A), spike detection (Fig. 1B), waveform alignment (Fig. 1C), and representation of the spike waveforms as points in a high-dimensional vector space (Fig. 1D)—are standard in nearly all spike sorting workflows. These steps yield a set of spike waveforms, each of which is represented by a vector in R^n (the n-dimensional real vector space), where n corresponds to the number of samples in each waveform and the vector components are the recorded samples.

The basic assumption underlying spike sorting is that waveforms from the same neuron are similar, while waveforms from different neurons exhibit distinctive features. Consequently, points in R^n corresponding to spikes from the same neuron should cluster together, whereas points corresponding to spikes from different neurons should remain separated in well distinguished clusters. The basic goal of a spike sorting algorithm is to efficiently identify the number and identity of such clusters in spike waveform data.

However, clustering in high-dimensional spaces is usually ineffective due to the "curse of dimensionality" (38), leading to sparse data and unreliable clustering. However, spike waveforms tend to lie on low-dimensional manifolds (39) over which clustering is feasible, which makes dimensionality-reduction a critical step in any spike-sorting pipeline. To the best of the authors' knowledge, all existing spike sorting methods perform dimensionality reduction either using linear projections (PCA, Wavelet decomposition) or fixed, expert-defined, nonlinear projections (Fig. 1E, F). Our approach, by contrast, employs UMAP– an unsupervised, nonlinear algorithm with a proven ability to preserve the essential topological and geometrical properties of the original data (23, 27, 40) – to reduce the dimensionality of spiking data (Fig. 1G-I).

A common challenge in applying dimensionality reduction to spiking data is that waveforms from different neurons may collapse into a single spurious cluster in the reduced space. Methods such as principal component analysis (PCA) (Fig. 1E, F), Wavelet decomposition (16), or expert-defined metrics (22), tend to struggle with complex, nonlinear boundaries between clusters. However, the existence of clusters within a dataset is a topological property that can be formalized using the concept of connected components of a topological manifold (41). UMAP-based dimensionality reduction can exploit the topological nature of clustering by provably preserving the topological features of the high-dimensional dataset (23, 24) in the low-dimensional projection. It does so in two steps:

1. Uniform Manifold Approximation (UMA), in which the nonlinear structure of the data is encoded into a fuzzy simplicial complex (Fig. 1G).
2. Projection (P), in which the fuzzy simplicial complex is optimally embedded into a low-dimensional space (Fig. 1H).

Because UMAP maintains clusters through its underlying topological framework, traditional clustering algorithms can effectively discern putative neurons within the projected data set (Fig. 1I). That is, the theoretical guarantees provided by UMAP enhance the prospects of splitting more neurons than feature-based approaches. Spike sorting using UMAP alone, for example, can split all three example neurons in Figures 1F and 1I.

For the clustering step, we employed HDBSCAN (37), an unsupervised algorithm that automatically determines the number of clusters, assess their statistical quality, and labels outliers as noise (pink beads in Fig. S1). Figure S1 (A-D) compares HDBSCAN's performance against K-means, and Gaussian Mixture models after UMAP-based dimensionality reduction. Both K-means and Gaussian Mixture require to specify in advance the number of clusters to be detected and lack mechanisms to label outliers as noise. Such conditions normally result in misclassification, particularly for non-convex cluster geometries (for example, electrode A illustrated in Fig. S1)—and they also make it harder to separate larger clusters (like electrode B in Fig. S1). In contrast, HDBSCAN's ability to isolate clusters while ignoring outliers provides greater accuracy and consistency in spike sorting results.

These results highlight the limitations of traditional clustering algorithms in identifying the best number of clusters, handling noisy data, and managing highly complex cluster shapes among data points. By integrating UMAP's non-linear dimensionality reduction

technique with HDBSCAN's adaptive clustering framework, the proposed methodology is a more effective and trustworthy approach to neuron separation. The subsequent sections illustrate the competency of this pipeline under varying situations.

Some useful spike sorting metrics

For the analysis presented in the following sections, we need metrics that can identify spuriously sorted spikes and quantify overall spike sorting performance (Fig. S2).

Consider two collections of spikes S_i and S_j obtained from the same experiment using different electrodes (recorded simultaneously) and associated with putative neurons i and j, respectively. The inclusion index $I_{i,j}$ is defined as the percentage of spikes in S_i that also appear in S_j (Fig. S2A-D). If both $I_{i,j}$ and $I_{j,i}$ are small, then S_i and S_j only share a few spikes, indicating that they likely come from different neurons. If $I_{i,j}$ is large but $I_{j,i}$ is small, then most of S_i is contained in S_j but not vice versa, which suggests that either S_j is contaminated by spikes from other neurons or that some spikes from the neuron identified by S_j were lost in S_i. Conversely, if both $I_{i,j}$ and $I_{j,i}$ are large, then S_i and S_j share most of their spikes and likely represent the same neuron.

A few datasets have one or several neurons that are identified as ground truth (GT). The set of spikes from a ground truth neuron is referred to as S_{GT}. In synthetic datasets, where the process used to create spike waveforms can be completely controlled, thus defining GT neurons is very straightforward. However, a few experimental spiking datasets possess GT neurons, which are usually derived from simultaneous intracellular and extracellular recordings enabling exact determination of the extracellular spikes from GT neurons.

When a GT neuron is available, with spike collection S_{GT}, we define the sorting quality $Q_{i,GT}$ of a sorted spike collection S_i with respect to S_{GT} as the average of $I_{i,GT}$ and $I_{GT,i}$, i.e., $Q_{i,GT} = {I_{i,GT} + I_{GT,i}}/{2}$ (Fig. S2A-D).

Only if both $I_{i,GT}$ and $I_{GT,i}$ are large does S_i faithfully represent S_{GT} (the Optimal case in Fig. S2D). If both are small (Unrelated case in Fig. S2A), then S_i and S_{GT} share only a few spikes. If $I_{i,GT}$ is small and $I_{GT,i}$ is large (the High Contamination case), S_i includes most GT spikes but is also contaminated by spikes from other neurons. If $I_{i,GT}$ is large and $I_{GT,i}$ is small (the High Loss case), S_i contains only a fraction of the GT neuron's spikes.

Finally, consider the case in which $n > 2$ nearby electrodes $E_1, \dots, E_n$ are used to sort the same set of neurons and let S_i be a collection of spikes sorted from electrode E_i. We can construct an inclusion index matrix M with off-diagonal elements $M_{i,j} = I_{i,j}$, $i \neq j$, and arbitrary diagonal elements (Fig. S2E). If one of the collections $S_k = S_{GT}$ corresponds to a GT neuron, it can be used as a reference to evaluate how accurately GT neuron spikes were sorted by other collections (matrices in Fig. S2A-D).

In the following sections, we will rely on the introduced notions of inclusion index, sorting quality, and inclusion index matrix to quantify spike sorting performance under a variety of experimental and synthetic conditions.

UMAP-based spike sorting achieves superior and robust performance

A common challenge in spike sorting arises when the distance between the recording electrode and the recorded neurons increases. Under these conditions, spike waveforms from different neurons begin to overlap, and background noise becomes more pronounced. To assess the robustness of UMAP-based spike sorting in such scenarios, we used a synthetic dataset created by the Quian Quiroga group (16).

Figure 2 compares the performance of the proposed UMAP-based method in sorting a complex mixture of spike waveforms from three synthetic GT neurons (inset Fig. 2A) against PCA- and Wavelet-based methods. In each case, HDBSCAN was used to cluster the low-dimensional projections. Notably, the 2D UMAP projection (Fig. 2A, left panel) sharply separates the waveforms into three distinct clusters that overlap well with the three GT neurons datapoints. Conversely, Wavelet (middle panel) and PCA (right panel) projections hardly reveal any well separated cluster. Although adding a third dimension (Fig. S3A for PCA and Fig. S3B for Wavelet) can sometimes improve separability, it also complicates visualization, making it harder for researchers to interpret the results—an issue that persists in other datasets (Fig. S3A & B, right panels). The UMAP method is not affected by these issues, by achieving clear and well sorted clustering of the datapoints in just two dimensions, thus preserving ease of visualization and interpretation. In addition, UMAP-based sorting quality (Fig. 2D) is remarkably insensitive to the used projection dimension ($Q_{UMAP,GT} = 0.83$). By contrast, the quality of Wavelet- ($Q_{WL,GT} = 0.68$) and PCA-based ($Q_{PCA,GT} = 0.58$) sorting exhibits noticeable fluctuation as the used projection dimension is varied, which makes it virtually impossible to determine for these two methods an optimal projection dimension. UMAP-based sorting quality is also

remarkably robust to more complex and challenging perturbations on the recorded waveforms, like drifting and bursting (Fig. S3C, left and right). In contrast, Wavelet- and PCA-based methods show significant performance declines when faced with the same perturbations.

Because noise is a critical factor in neural spike sorting, we explored how UMAP handles increasing noise levels (η ranging from 0.05 to 0.2) added to three synthetic spike waveforms from (16) (Fig. S3D), and compare it against PCA and Wavelet decomposition (Fig. S3E, right, middle, and left panels, respectively). Again, UMAP-based sorting is remarkably stable across a wide range of projection dimensions, and it systematically outperforms both PCA- and Wavelet-based sorting as η increases. Even at the highest noise level ($\eta = 0.2$), UMAP achieves over 70% sorting performance. Finally, to test the ability of UMAP-based sorting to detect "silent" neurons, we randomly removed an increasing fraction of the datapoints from one of the synthetic GT neuron clusters (the red one, as per Fig. 2A). Remarkably, UMAP still ensured almost perfect (~1) sorting quality when only 40% of the cluster datapoints were retained and still very good (~0.8) sorting quality even when only 20% of the datapoints were retained. This highlights UMAP ability to effectively isolated spikes from neurons with low firing rates, which are often key encoders of a task parameters due to their low variability (31, 42). Conversely, the already suboptimal sorting quality of both PCA- and Wavelet-based methods dropped much more rapidly as the fraction of removed spikes was increased. Together with its robustness to noise, these findings highlight UMAP's ability to reliably sort low-firing neurons, which suggests it as a powerful spike sorting tool in challenging experimental conditions.

The Cost of Overlooking Low-Firing-Rate Neurons in Neural Spike Sorting

In Figures 3, S4, and S5, we illustrate how merging spikes from different neurons into a single spurious multi-unit entity can severely distort and be detrimental to the analysis of neural information encoding. First and foremost, each multi-unit entity created by merging spikes from distinct neurons inevitably exhibits a (potentially much) higher firing rate than the individual units, i.e., the sum of their firing rates. As a result, the average firing rate can be exaggerated, and this can cause significant errors in later analyses. Moreover, neurons with low firing rates are particularly likely to be missed. Their few spikes can be lost in high-firing-rate multi-unit clusters unless appropriate dimensionality reduction techniques are used to retain their distinctiveness. Therefore,

the potentially important information signaled by low-firing-rate neurons can be buried in the noise of high-firing-rate multi-unit clusters. Figures 3A, S4A, and S5A sketch this fundamental problem. While two isolated units might show clear and distinct task parameter and condition encoding (as sketched by well separated light to dark blue firing rate traces), merging them into a single multi-unit entity can average out and thus miss such encoding (as sketched by overlapping rate traces). This simple phenomenon might erroneously lead to discard the spurious multi-unit activity spikes as irrelevant, thus losing potentially precious and important pieces of neural code.

Figures 3B-E, S4B-E, and S5B-E provide different experimental examples of the detrimental merging of two single units into a spurious one. Figures 3B and S4B illustrate the course of a time interval comparison task (TICT) (43, 44) in which animals compare the duration of two intervals (Int1 and Int2, ranging from 400 ms and 2000 ms) and report which one is longer. During the 2-second working memory period between the intervals, the animal must retain Int1-related information in order to compare it with Int2. The anatomical diagram in the same panel shows the location the dorsal premotor cortex (DPC), where the neurons in Figures 3C-E and S4C-E were recorded from. Figures 3C and S4C show the raster plots (top) and firing rates (middle), aligned to the start of the delay period, of a multi-unit entity created by merging two putative neurons identified through the UMAP-based pipeline but that were combined in a single (higher firing rate) neuron when the PCA-based method was used. The activity of the two individual neurons is shown in Figures 3D, 3E, S4D and S4E. Notably, the putative neuron in Figure 3D has a relatively lower firing rate but carries significantly more information about Int1 during the delay period, as revealed by mutual information analyses (bottom panels in Figs 3C-E & S4C-E). The multi-unit activity in Figure 3C carries significantly less information than the combined values of the two isolated neurons, which provides evidence and an experimental example of how merging different neurons into spurious multi-units can erase potentially critical information carried by low-firing-rate neurons. Although the putative neuron in Figure S4D is almost silent most of the time —which is probably a key reason for why it is likely to be missed during sorting—it carries more information about Int1 at the onset of the delay period than both the multi-unit entity in Figure S4C and the other putative neuron in Figure S4E.

Figure S5 shows a similar effect in a different task and brain area. Figure S5B outlines the course a tactile detection task (45, 46) and shows the location of the ventral premotor cortex (VPC), where the neurons in Figures S5C-E were recorded from. In this task, the

animals reports whether a vibrotactile stimulus (0-24 μm in amplitude) is present or not. Figure S5C displays the multi-unit activity obtained by merging the spikes of the two putative neurons in Figures S5D and S5E. The two neurons in Figures S5D and S5E differ substantially in both firing rates and encoding dynamics: the neuron in panel D persistently maintain a stable encoding during the delay period, while the neuron in panel E responds strongly, but only transiently, to stimulus presentation and its activity rapidly decreases during the delay period. Merging these two putative neurons completely conceals their distinctly different encoding profiles–most critically, it erases the persistent encoding of the lower-firing-rate neuron (D) and diminishes the nuanced, non-monotonic delay activity of the higher-firing-rate neuron (E).

The experimental evidence provided in this section shows that dimensionality reduction methods that are sensitive to data-point density–such as PCA–likely fail to detect and isolate the neurons with lowest firing rates. Their spikes tend indeed to be subsumed into multi-unit clusters dominated by higher-firing-rate neurons. By drawing on topological and geometric principles, UMAP-based spike sorting addresses this issue effectively, preserving the unique contribution of low-firing-rate neurons and safeguarding against the loss of valuable information.

UMAP-Based spike sorting improves neural information encoding analysis by enhancing low-firing-rate neurons detection

In this section, we apply UMAP-based sorting to recordings from three brain areas–dorsal premotor cortex (DPC), ventral premotor cortex (VPC), and second somatosensory cortex (S2)–while animals perform the TICT (43) outlined in Figures 3B and S4B. Our objective is to show that isolating low-firing-rate neurons through effective spike sorting not only increases the total number of identified cells but also boosts the amount of information extracted from the data. Figures 3, S4, and S5 report examples of this phenomenon. Figure 4 reveals that the same phenomenon is much more general.

Figure 4A shows the firing-rate cumulative probability distribution of the putative neural population sorted using either UMAP-based or PCA-based methods from DPC, VPC and S2 (brain views on Fig. 4B) recordings. In every dataset, UMAP-based sorting consistently identifies more lower-firing-rate than PCA-based sorting.

When the same analysis was performed on data sorted from only one of three areas—S2 (Fig. 4C), VPC (Fig. 4D), and DPC (Fig. 4E)—it confirmed that UMAP-based sorting consistently detects a larger population of low-firing-rate neurons across all regions (left

panels). Furthermore, in all areas UMAP-based sorting significantly increased the total number of simultaneously recorded neurons (central panels). This increment in simultaneously-recorded neurons count is critical for exploring pairwise spike correlations (1–3). For instance, in DPC, the number of simultaneously recorded neurons rises from roughly 8 to nearly 13, thereby increasing the average number of neuron pairs from about 28 to almost 70. Finally, the mutual information extracted from neuronal spikes about task parameters was consistently higher when using UMAP-based sorting (right panels). This observation is consistent with the enhanced clarity and cluster separation achieved by UMAP-based sorting highlighted in Figure 3. In short, UMAP-based sorting not only identifies more neurons—particularly those with low firing rates—but also enhances the overall information decoded from those cells.

Comparing UMAP-based sorting and SpyKING CIRCUS in multielectrode array recordings: the role of distance.

In this section we explore the performance and potential of UMAP-based spike sorting on multielectrode array (MEA) recordings and compare it with SpyKING CIRCUS (13), a widely used, high-performing software specifically designed for *in vivo* and *in vitro* MEA recordings. The performance of the two sorting methods was quantified on the same dataset as the original study (13), which includes extracellular spike recordings from a surface MEA, as well as intracellular recordings that define a GT neuron.

Thanks to the regular grid spatial arrangement of MEA, it is straightforward to quantify the distance between each electrode and the GT neuron. We leverage this fact to analyze how sorting performance varies with electrode-to-neuron distance for both UMAP-based sorting and SpyKING CIRCUS.

We focused on a square 5x5 electrode grid centered at the GT neuron (Fig. 5A-D). The position of each electrode in the grid relative to the GT neuron is represented by a coordinate (i, j), $i, j = -2, \dots, 0 \dots, 2$, with (0,0) corresponding to the electrode closer to the GT neuron. Figure 5A shows a representative MEA recording where UMAP-based sorting not only achieves near-perfect performance at the central electrode (closest to the GT neuron) but also maintains good sorting quality at most surrounding electrodes. Although more GT neuron spikes are lost and contamination increases with distance, UMAP-based sorting is ablet to correctly identify a substantial number of GT neuron spikes across a wide range of electrodes. This observation is confirmed in Figure 5B, which shows the robust distance-dependent performance of UMAP-based sorting

averaged over 16 MEA experiments (13). In contrast, sorting of the same data as Fig. 5A with SpyKING CIRCUS (Fig. 5C) reaches only suboptimal performance already at the central electrode and the sorting quality drops abruptly to zero in all surrounding electrode. The extreme sensitivity of SpyKING CIRCUS to electrode-neuron distance is confirmed by the multi-experiment averaged data in Figure 5D. Figures 5E and 5F summarize these observations by grouping electrodes with similar distance from the GT neuron and plotting how sorting quality changes with distance. While SpyKING CIRCUS performance sharply falls to zero at the nearest neighboring electrodes, UMAP-based sorting decreases more slowly and remains significantly above zero up to 40 µm away. Further note that SpyKING CIRCUS performance at the central electrode is comparable to UMAP-based sorting performance at electrodes roughly 30 µm from the GT neuron. An exponential fit of the data in Figure 5F offers a more quantitative view of the spatial decay in sorting performance of the two methods. As the recording electrode is moved away from the recorded neuron, UMAP-based sorting performance decays smoothly and with a spatial constant that is roughly two orders of magnitude larger than that obtained with SpyKING CIRCUS. Besides being robust to heterogeneous firing rates and to various sources of noise, UMAP-based sorting is also robust to the effects of a non-negligeable distance between electrode and neuron.

Figure S6, further dissects the sorting quality ($Q_{i,GT}$) into two probabilities introduced earlier (Fig. S2). $I_{i,GT}$, the probability that a spike from S_i appears in S_{GT}, and $I_{GT,i}$, the probability that a spike from S_{GT} appears in S_i. Both probabilities exhibit distance-dependent decay similar to $Q_{i,GT}$. However, $I_{GT,i}$ drops more sharply than $I_{i,GT}$, indicating that lost GT spikes–rather than contamination from spikes of other neurons–primarily drive the overall decrease in sorting quality.

Comparing UMAP-based and Feature-based spike sorting in multi-electrode recordings

In Figure 6, we applied UMAP-based sorting to extracellular multi-electrode recordings obtained in Buzsaki's lab (33). Also in this database, an intracellular electrode was used to define a ground truth (GT) neuron whose extracellularly recorded spike waveforms are collected in S_{GT}. Simultaneously, four extracellular electrodes were placed at varying distances from the GT neuron (Fig. 6E). UMAP-based sorting of electrode 1 data revealed four well-separated clusters (Fig. 6A) associated with the waveforms in Fig. 6D. One of

the clusters almost perfectly overlap with S_{GT}, indicating successful identification of the GT neuron. The other three clusters' points to three other putative neurons. Figures 6H and 6K show the same data points as Fig. 6A and colored according to the same UMAP-based clustering, but projected using Wavelet decomposition and PCA, respectively. As detailed below, it is already clear from these projections that neither Wavelet- nor PCA-based sorting are able to identify and sort the GT neuron. In Figure 6F, data points that are shared with Figure 6A (i.e., spikes recorded on both electrode 1 and 3) are colored according to the same cluster colors as Figure 6A, whereas data points associated with spikes recorded on electrode 3 but not electrode 1 are drawn in purple). Notice that electrode E3 is farther from the GT neuron than E1 (Fig. 6E), making it more difficult to sort spikes from the GT neuron, in agreement with Fig. 5.

The 1st line of Figures 6B, 6I, and 6L, shows the GT neuron raster plot, drawn in green, while the (i+1)-th line, $i = 1, \dots, 4$, shows the raster plot of the spike cluster S_i sorted from electrode E_i (either through UMAP-, Wavelet-, or PCA-based methods, as indicated in the panel titles) that most closely matches (in the sense of maximizing the sorting quality $Q_{i,GT} = \frac{I_{i,GT}+I_{GT,i}}{2}$) the GT neuron spike cluster S_{GT}. Observe that each raster plot is horizontally split in two halves distinguished by different tones of blue that indicate (with the convention that higher, resp. lower, values are indicated by darker, resp. lighter, tones) the associated cluster inclusion indices $I_{i,GT}$ (top half color) and $I_{GT,i}$ (bottom half color. Hence, only the raster plots of spike clusters with a high sorting quality $Q_{i,GT}$ are drawn in a roughly uniform dark blue tone,

The cluster obtained from electrode E_3 in Figure 6B using UMAP-based sorting exhibits a low $I_{3,GT}$ but a high $I_{GT,3}$, suggesting contamination with non-GT spikes (Fig. S2C). Conversely, the cluster obtained from electrode E_4 exhibits a high $I_{4,GT}$ but a low $I_{GT,4}$, indicating a significant loss of GT neuron spikes. By contrast, the two clusters obtained from electrodes E_1 and E_2 are characterized by a high $I_{1,GT}$ and a high $I_{GT,1}$, thus identifying the GT neuron spikes with high quality (Fig. 6A). Figure 6C shows the remaining elements of the inclusion index matrix ($M_{i,j}$) that can be used to investigate inclusion relationships between the spike clusters sorted from different electrodes (see Fig. S2).

Figures 6H–6M shows the result of the same analysis but using Wavelet decomposition and PCA, respectively, instead of UMAP for the dimensionality reduction step. As shown in Figure 6H, when only the two largest Wavelet coefficients are used to project the data,

all spike waveforms are merged into a single cluster. In spite of this, the utilization of the six most considerable coefficients (illustrated in Figs. 6I and 6J) enhances the separation process; however, none of the clusters derived from the Wavelet method as precisely resembles S_{GT} as those derived through UMAP-based sorting. The cluster obtained from electrode E_1 is of the best quality of sorting, although it undergoes a comparatively high loss rate. In contrast, the clusters obtained from electrodes E_2, E_3, E_4 are of inferior qualities of sorting because they experience a high contamination rate. PCA-based sorting is worse and consistently produces big, highly contaminated clusters of high-firing frequency multi-unit activity for all electrodes.

In summary, only UMAP-based spike sorting reliably separates GT neuron spikes from at least two electrodes. Wavelet-based and PCA-based methods either merged multiple neurons' spikes (high contamination) or failed to capture a significant fraction of the GT neuron's spikes (high loss). Nonetheless, all methods—including UMAP—face increasing challenges as the electrode-to-neuron distance grows (Fig. 5).

DISCUSSION

Spike sorting, i.e., the identification and classification of single neuronal spikes from extracellular electrophysiological recordings, is essential to understanding neural codes. Conventional approaches typically use thresholding techniques in conjunction with linear dimensionality reduction methods, e.g., principal component analysis (PCA) or Wavelet decomposition, followed by clustering (16, 17, 47). Although effective for high-amplitude or regularly firing neurons, such linear approaches are ineffective at separating clusters that are divided by faint, nonlinear waveform disparities, particularly for large-scale recordings, thus leading to neuron misclassification and manual sorting of large quantities (6). In contrast, nonlinear dimensionality reduction methods—especially UMAP—provide mathematically rigorous embeddings that preserve both global and local topological relationships within spike waveform data (23, 24).

While analyzing massive datasets without spike sorting has recently gained popularity for exploring population-level latent dynamics (48, 49), critical single-neuron analyses can be severely compromised if sorting is omitted (50). Vital insight, for example derived from the relationship between noise correlation and signal correlation, depends to a large extent on properly sorted single-neuron data (1–3). Moreover, signal correlation must be orthogonal to noise correlation ~~in order~~ to achieve maximum population-encoded

information (3, 51). Neglecting spike sorting ~~neglect thus~~ risks losing valuable information about single-neuron dynamics, obscuring fundamental properties of neural computation. Our spike-sorting pipeline directly addresses these challenges, leveraging UMAP's nonlinear embeddings to successfully separate clusters based on subtle waveform differences with high precision, outperforming linear methods by a large margin in a range of experimental conditions (Figs. 2–4). This capacity not only reduces manual curation but also ensures complete neuron detection, including silent or low-firing units typically omitted.

One important subset of neurons that is usually neglected consists of low-firing or "silent" neurons (31, 42). Conventional approaches are typically not able to identify these neurons because of the low firing rates and low spike amplitudes, which are difficult to discriminate using variance-based clustering (10, 52). In contrast, UMAP's robustness to variability in spike densities provides strong detection and classification of silent neurons. Empirical results indicate that UMAP obtains a very high sorting accuracy even under extreme spike loss of up to 80% and that linear methods suffer a drastic reduction in performance. Thus, the application of UMAP for sorting facilitates the detection of more diverse and accurate neuronal populations, which greatly simplifies subsequent neural coding analysis and interpretation in cognitive neuroscience research (1–3, 51). Further, by maximizing both the number and diversity of the detected neurons, UMAP substantially improves studies on neural information encoding. Low-firing neurons, which are conventionally disregarded by default procedures, carry valuable, stable information with finely graded response variability (3, 32, 51). Our data for cognitive tasks demonstrate that omission of such neurons substantially distorts firing rate estimates and mutual information estimates of task-relevant variables. By accurately preserving low-firing neurons, UMAP yields higher quality sorted datasets and more valid interpretations of population codes, noise correlations, and task-related neural dynamics (6, 7, 10).

Another significant advantage of applying UMAP to spike sorting is its greater ability to preserve the spatial and temporal features contained in multi-electrode array (MEA) recordings. Contrary to traditional linear techniques that tend to exhibit significant decline in sorting capability with the increase in the distance between neurons and electrodes (as evident in Figure 5), UMAP is characterized by a smooth, gradual decline in performance. Such ~~a~~ characteristic helps in identifying an enormous number of genuine spikes even

where electrodes are placed at extremely large distances. The inclusion of spatial resolution improves the validity of neuronal identification across the array while preserving the positional relationships of neural activity necessary for the detection of functional connectivity (53–56). In addition, UMAP's insensitivity to temporal fluctuations, such as drifting and bursting, is one of the main determinants of its greater stability across long-term recordings, allowing for more stable monitoring of neuronal activity over time. These spatiotemporal advantages are particularly important for closed-loop experimental paradigms and real-time applications, where stable, high-quality neural representations need to be maintained to facilitate precise decoding and adaptive intervention (57–59).

Computational scalability is yet another key benefit of our UMAP-based pipeline, essential for modern high-density electrophysiological experiments with thousands of neurons recorded simultaneously (6, 9). In contrast to methods like t-SNE or ISOMAP, which become intractable with large datasets, UMAP scales roughly linearly with dataset size (23, 24). Our results consistently demonstrate UMAP's higher performance compared to linear methods, making it especially beneficial for real-time or near-real-time spike sorting in closed-loop experimental paradigms. Observably, the stable nonlinear embedding of UMAP, combined with the density-based clustering of HDBSCAN, enables an end-to-end fully automated spike sorting pipeline. Unlike clustering algorithms such as Gaussian mixture models or K-means, which require pre-defined cluster numbers and struggle with non-convex cluster borders, UMAP-HDBSCAN automatically determines optimal clusters and considers outliers as noise. This automation significantly reduces and accelerates the spike sorting pipeline, with minimal manual intervention and maximal reproducibility.

Relative to other contemporary spike sorting approaches with UMAP, like WaveMAP (28), which involves graph-based clustering, or P-sort (60), which involves PCA preprocessing specialized for cerebellar data, our method utilizes only UMAP along with HDBSCAN clustering. This direct and efficient method obviates additional preprocessing or specialized clustering algorithms, thus enhancing computational efficiency, enabling automation, and increasing generalizability to various types of neurons and experimental conditions. Consequently, our method effectively captures a wider array of neurons, substantially improving analyses of population-level neural coding and theoretical insights into neural computation. Future studies could further validate our method’s broad

applicability, potentially integrating deep learning techniques for a more powerful, comprehensive unsupervised sorting pipeline.

ACKNOWLEDGMENTS

This work was supported by grants PAPIIT-IN210819, PAPIIT-IN205022 and IN203825 (to R.R.-P.) from the Dirección de Asuntos del Personal Académico de la Universidad Nacional Autónoma de México and CONAHCYT-319347 (to R.R.-P.) from Consejo Nacional de Ciencia y Tecnología; IBRO Early Career Award 2022 (to R.R.-P.) from International Brain Research Association. L.B. is a postdoctoral researcher (Postdoctoral fellowship CONACYT-838783).

Methods

Full pipeline's code and demo data are available on github: https://github.com/d-suarez-b/Nonlinear-Dimensionality-Reduction-for-Spike-Sorting-Analysis

Spike sorting workflow

Our spike sorting procedure has three main steps: data preprocessing, feature extraction, and clustering. The innovation in our method lies in improving feature extraction by nonlinear dimensionality reduction of the spike waveforms and hierarchical clustering for classification as neural units.

Data preprocessing

To minimize the noise within the waveforms, the third-order Savitzky-Golay filter was used, with a window of 5 samples. This filter effectively smooths the signal while preserving essential spike features. After filtering, waveforms are interpolated using a Piecewise Cubic Hermite Interpolating Polynomial (PCHIP), which maintains a smooth interpolation and preserves the original waveform shape. Each spike is then aligned at its minimum, followed by downsampling to return to the original point count while preserving alignment.

Feature extraction

Following preprocessing, we applied UMAP (Uniform Manifold Approximation and Projection) (23) to achieve nonlinear dimensional reduction of the waveforms, using hyperparameters set as: n_components=2, min_dist=0, and n_neighbors=13. To gauge performance in terms of dimensionality, we established a grid that was purposed for hyperparameter tuning. This included searching the n_neighbors parameter while maintaining min_dist at 0, thereby ensuring maximum fidelity of density in the low-

dimensional space. We then employed the Kolmogorov-Smirnov test to calculate the divergence between the normal distribution and the distribution of data in the low-dimensional space. The hyperparameters with the highest divergence were finally established. UMAP utilizes mathematical principles based on graph theory and differential geometry, thus allowing it to capture local and global arrangements of the data well. In comparison to other approaches, like t-SNE, UMAP offers improved performance in terms of global structure preservation, thus being particularly beneficial for large multidimensional data sets (23). By maintaining nonlinear relationships between the data, UMAP offers a faithful low-dimensional representation of spike waveforms.

Clustering

On the UMAP-reduced data, we utilized HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise) (37) for clustering. HDBSCAN creates a dendrogram of clusters by point connectivity with the help of a minimum distance parameter. Through condensation of the dendrogram, HDBSCAN identifies stable clusters, traveling down the hierarchy until an optimum density threshold is met, and selecting the most prominent clusters. Noise is natively handled in HDBSCAN by labeling points outside of dense regions as noise, and so it is an ideal algorithm to pair with UMAP for spike sorting. This low-dimensional subspace from UMAP also enables manual visualization and curation, allowing further validation by checking unit stability, ISI violations, and waveform consistency.

Inclusion Index Matrix

To compute the inclusion matrix $M_{i,j}$ (Fig. S2), we evaluated the percentage of spikes in raster plot i that also appear in raster plot j ($I_{i,j}$). $M_{i,j}$ is a directional index that quantifies the overlap and similarity between spike trains. As shown in Figs. 6 and S2, this matrix is not symmetric, reflecting the directional nature of the comparison.

In cases where spike times belong to the ground truth (GT) (S_{GT}), it is possible to establish the number of matched, missed, and false-positive spikes with respect to a cluster k (S_k):

1. Matching Spikes: $n_{match}^{k} = \{|t_j^k - s_i| < \varepsilon\}$
2. Missed Spikes: $n_{miss}^{k} := n_{GT} - n_{match}^{k}$
3. False Positives: $n_{fp}^{k} := n_k - n_{match}^{k}$

Using these definitions, we calculated the precision p_k ($I_{k,GT}$), which decrease when S_k is contaminated with spikes from other neurons, and the recall r_k ($I_{GT,k}$), which decrease when spikes from S_{GT} are lost in S_k:

- $I_{k,GT}$: Precision (contamination): $p_k := \frac{n_{match}^k}{n_{match}^k + n_{fp}^k}$
- $I_{GT,k}$: Recall (lost): $r_k := \frac{n_{match}^k}{n_{match}^k + n_{miss}^k}$

Note that these probabilities correspond to the GT column ($M_{i,GT}$) and the GT row ($M_{GT,i}$) in the inclusion matrices displayed in Fig. 6. As shown in Figure S2, when $I_{k,GT}$ (p_k) is low, contamination is high, corresponding to element (2,1). This indicates that a large number of spikes identified in the cluster from E1 do not belong to the GT neuron. Conversely, in Figure S2, when $I_{GT,k}$ is low, losses are high, corresponding to element (1,2). If a GT neuron is available, then we can define the sorting quality $Q_{i,GT}$ of a collection of sorted spikes S_i with respect to S_{GT} as the average of the inclusion index $I_{i,GT}$ of S_i in S_{GT} and the inclusion index $I_{GT,i}$ of S_{GT} in S_i:

$$Q_{i,GT} = \frac{I_{i,GT} + I_{GT,i}}{2}$$

The other elements of the matrix $M_{i,j}$ quantify the inclusion probabilities between the different clusters identified. For instance, the element (E4, E3) from $I_{E4,E3}$ represents the probability that a spike from the cluster identified in E4 (S_{E4}) is also present in the cluster identified in E3 (S_{E3}).

Comparative analysis on different databases

The "Wave_Clus" (16) comprises 594 distinct average spike waveforms, derived from actual recordings, which served as templates for generating synthetic signals. Randomly chosen spikes were added at arbitrary times and with varying amplitudes to replicate background noise. Simulation of various signal-to-noise ratio (SNR) conditions was accomplished by changing the ratio of the signal amplitude to noise amplitude. More details about the development of this database are given in reference (16).

We employed four large datasets in our study: C_Easy1, C_Easy2, C_Difficult1, and C_Difficult2. C_Easy1 had eight noise levels ranging from 0.05 to 0.4 with increments of 0.05, while the other three datasets included four noise levels ranging from 0.05 to 0.2. "Easy" and "difficult" describe the level of overlap of spikes in each dataset. To test the effectiveness of our method in complicated scenarios, we used datasets created with bursty neuronal activity and drifting dynamics. The results were then compared with the

ones obtained from wavelet- and PCA-based methods. Spikes were detected from continuous recordings using the ground-truth spike times available in the database, providing labeled samples that are very suitable for testing our supervised deep learning model. Each spike contained 64 sampling points, and the database was sampled at a frequency of 24 kHz.

In addition, the Buzsaki dataset (33) consists of the extracellular and juxtaposed intracellular recordings of thirty Sprague-Dawley anaesthetized rats. Electrical activity recordings were done at the hippocampus CA1 pyramidal layer.

For rodent hippocampal data (33) and synthetic data (16) (Figs. 2 and 6), we compared UMAP with PCA and Wavelets, using the optimal number of PCA components and the optimal number of wavelet features. HDBSCAN clustering parameters were kept consistent across tests. The highest sorting quality ($Q_{i,GT}$) value cluster was used for comparison with the ground truth (GT). For Figure 2, only spikes corresponding to a single synthetic neuron were treated as ground truth.

Evaluating spike sorting performance

To assess the accuracy of different spike sorting algorithms, we used datasets with known ground truth (GT) and calculated performance metrics in terms of inclusion indexes, losses and contamination probabilities (see Fig. S2).

Spatial accuracy

For an in vitro mouse retinal MEA (256 channels), UMAP-based sorting was compared to Spyking Circus (Fig. 5), using database-specific parameters (13). To assess spatial accuracy, sorting quality ($Q_{i,GT}$) for each electrode was fitted to a decaying exponential as a function of distance from the reference electrode (highest average $Q_{i,GT}$ cluster among 256 electrodes).

Databases without Ground Truth

For the tetrode recording database from rhesus monkeys (Figs. 3, 4), we applied identical data preprocessing to both UMAP and PCA feature extraction algorithms. For clustering, HDBScann was implemented (Fig. S1). The average firing rate was computed for each neuron during the interval comparison task. The average firing rate was calculated across all intervals trials, throughout the entire task duration (43). We employed recording from secondary somatosensory area (S2), from ventral premotor cortex (VPC) and from dorsal premotor cortex (DPC) (43, 46).

In order to compare the distributions of the mean firing rates, we constructed cumulative distribution functions (CDFs) of those neurons sorted by UMAP and PCA within each recording area. In addition, for each recording, we compared the number of neurons that each method detected and computed the respective mutual information from these comparisons.

Principal Component Analysis (PCA) sorting

The key point of performing Principal Component Analysis (PCA) on spike waveform data for spike sorting is to discover an alternate coordinate system wherein the waveforms can be expressed in a more compact and reduced form. That is, the objective is to construct a low-dimensional subspace that contains most of the variance present in the high-dimensional waveform space. In general, there is a dramatic reduction in the number of significant dimensions, from a number equal to the number of sample points per waveform (e.g., several hundreds) to only a very small number of principal components (PCs) that account for the majority of the variance. Principal Component Analysis (PCA) generates another coordinate system for the high-dimensional data, where the first principal component accounts for the largest amount of variance relative to the shapes of the waveforms. The second principal component accounts for the next largest portion of the variance, but each subsequent axis must be constrained to be orthogonal to all the previous axes.

PCA is derived from the covariance matrix of the waveform data, which is estimated across all waveforms and time points. The covariance matrix is provided by:

$$Cov(W_i\,,\,W_j) = \frac{\sum_{k=1}^{N}\left(w_i^k - \overline{w_i}\right)\left(w_j^k - \overline{w_j}\right)}{N-1} \qquad \text{(Eq. S1)}$$

where N is the total number of waveforms (spikes) being considered, w_i^k is the amplitude of k-th waveform at i-th time point. $\overline{w_j}$ is the mean amplitude at time point j across all waveforms:

$$\overline{w_i} = \frac{1}{N}\sum_{k=1}^{N} w_i^k \qquad \text{(Eq. S2)}$$

The diagonalization of the covariance matrix $Cov(W_i\,,\,W_j)$, results in a new coordinate system represented by the columns of matrix V, and the columns are referred to as the derived axes or principal components (PCs). Λ is also a diagonal matrix with positive entries, where the diagonal entries of Λ represent the amount of variance in the waveform data captured by the corresponding principal components. Then we arrange the primary

components based on how much variance they hold. The *k-th* principal component projection of *l-th* waveform data is represented as:

$$\widehat{w_l} = \sum_{i=1}^{T} w_{li}\, v_{ik} \qquad \text{(Eq. S3)}$$

where T is the number of time points in each waveform, v_{ik} is the i-th element of the k-th PC, and $\widehat{w_l}$ is the projection of waveform w_l onto k-th principal component. Therefore, the PCs are linear combinations of amplitudes at different time points within the waveform. The contribution of each time points to a given $\widehat{w_l}$ is represented by the i-th element of the k-th column of eigenvectors matrix V. These PCs provide a low-dimensional description of the waveform data in this coding subspace, facilitating more effective spike sorting by highlighting the most significant variations in waveform shapes.

Wavelet sorting

The primary motivation behind the application of the wavelet transform in spike waveform data analysis for spike sorting is to obtain informative features that represent the waveforms in a more compact and concise way for later classification. The wavelet transform depicts each spike waveform as a collection of coefficients that embody both temporal and frequency details, thus enabling a close study of waveform configurations. By knowingly selecting a certain subset of important wavelet coefficients, we effectively decrease the data dimensionality from the original number of time points with which each waveform is described to a lower, more compact set of features that include the most relevant discriminative information.

The continuous wavelet transforms (CWT), or discrete wavelet transform (DWT) is applied to each waveform. The wavelet transform of a waveform is defined as:

$$W(a,\, b) = \frac{1}{\sqrt{a}} \int x(t)\overline{\psi}\left(\frac{t-b}{a}\right) dt \qquad \text{(Eq. S4)}$$

where, a>0 is the wavelet coefficient at scale. This coefficient represents the dilation, while and b the time shift or translation. Finally, $\overline{\psi}$, is the complex conjugate of the mother wavelet function ψ. In practice, we use the discrete wavelet transform (DWT) due to computational efficiency and data discretization. The DWT decomposes the waveform into approximation and detail coefficients at various levels of decomposition. Mathematically, the DWT coefficients are computed using a series of high-pass and low-pass filters followed by downsampling.

Approximation coefficients at level:

$$T_{mn} = \int_{-\infty}^{\infty} x(t)\psi_{mn}(t)dt \qquad \text{(Eq. S5)}$$

Detail coefficients at level j:

$$S_{jn} = \int_{-\infty}^{\infty} x(t)\phi_{jn}(t)dt \qquad \text{(Eq. S6)}$$

After decomposing the waveform into multiple levels, we select a subset of significant wavelet coefficients that capture the most relevant features for spike classification. This selection can be based on criteria such as the largest absolute values or statistical measures of variance across the dataset. The selected wavelet coefficients form a feature vector for each waveform:

$$d_m(t) = \sum_{n=-\infty}^{\infty} T_{mn}\,\psi_{mn}(t) \qquad \text{(Eq. S7)}$$

where m is the number of selected coefficients and d_m are the detail coefficients at specific levels m and positions t. These feature vectors provide a low-dimensional representation of the spike waveforms in a feature space suitable for clustering. Clustering algorithms (e.g., K-means, Gaussian Mixture) are then applied to the feature vectors to group similar spikes together, effectively sorting neurons based on the waveform shapes. Unlike Principal Component Analysis, wavelet decomposition does not inherently rank its coefficients. To establish a ranking, we applied a previously proposed method (16). Briefly, the Kolmogorov-Smirnov test was used to evaluate the distance between each coefficient's distribution and the normal distribution, with larger distances indicating greater divergence. The coefficients were then ranked accordingly, and the K coefficients with the highest values were selected for further analysis.

Data Analysis

Firing Rate. For each neuron, a time-dependent firing rate was computed on a trial-by-trial basis using overlapping, rectangular, causal windows of 200 ms in length, with a step size of 20 ms.

Mutual Information. For each neuron, we quantified the relationship between the firing rate ($r(t)$) and the presented stimuli s in each time window using Shannon mutual information (61, 62). This measure captures nonlinear dependencies between the two variables. To compute $I(r;s)$, we used the conditional firing rate distributions across

stimulus conditions ($P(r|s)$) along with the overall firing rate distribution ($P(r)$), obtained by pooling trials across all interval values:

$$I(r\,;\,s) = \sum_{i}^{N}\sum_{k}^{M} P(r_i|s_k)P(s_k)\log_2\left(\frac{P(r_i|s_k)}{P(r_i)}\right) \qquad \text{(Eq. S8)}$$

The mutual information significance was tested for each neuron for the given time intervals. We calculated a permuted mutual information value by conducting 1000 permutations of the trials. A mutual information value was deemed significant if the probability that a permutation yielded an equal or higher value was lower than 0.05 ($p < 0.05$). Additionally, to account for the problem raised by finite sampling, we utilized the correction established in (63). At the time of significance testing, we also used a multiple comparison correction by implementing a clustering method that has been described before (64). This involved keeping only the group of time bins that had significant connectivity and a predefined size.

REFERENCES

1. M. Valente, *et al.*, Correlations enhance the behavioral readout of neural population activity in association cortex. *Nat. Neurosci.* **24**, 975–986 (2021).
2. S. Panzeri, M. Moroni, H. Safaai, C. D. Harvey, The structures and functions of correlations in neural population codes. *Nat. Rev. Neurosci.* **23**, 551–567 (2022).
3. O. I. Rumyantsev, *et al.*, Fundamental bounds on the fidelity of sensory cortical coding. *Nature* **580**, 100–105 (2020).
4. A. Tauste Campo, *et al.*, Feed-forward information and zero-lag synchronization in the sensory thalamocortical circuit are modulated during stimulus perception. *PNAS* **116**, 7513–7522 (2019).
5. J. D. Semedo, A. Zandvakili, C. K. Machens, B. M. Yu, A. Kohn, Cortical Areas Interact through a Communication Subspace. *Neuron* **102**, 249-259.e4 (2019).
6. J. J. Jun, *et al.*, Fully integrated silicon probes for high-density recording of neural activity. *Nature* **551**, 232–236 (2017).
7. N. A. Steinmetz, *et al.*, Neuropixels 2.0: A miniaturized high-density probe for stable, long-term brain recordings. *Science* **372**, eabf4588 (2021).
8. G. Hong, C. M. Lieber, Novel electrode technologies for neural recordings. *Nat. Rev. Neurosci.* **20**, 330–345 (2019).
9. Z. Zhao, *et al.*, Ultraflexible electrode arrays for months-long high-density

electrophysiological mapping of thousands of neurons in rodents. *Nat. Biomed. Eng.* (2022). https://doi.org/10.1038/s41551-022-00941-y.

10. G. Hilgen, *et al.*, Unsupervised Spike Sorting for Large-Scale, High-Density Multielectrode Arrays. *Cell Rep.* **18**, 2521–2532 (2017).
11. C. Rossant, *et al.*, Spike sorting for large, dense electrode arrays. *Nat. Neurosci.* **19**, 634–641 (2016).
12. J. H. Lee, *et al.*, YASS: yet another spike sorter. *Adv. Neural Inf. Process. Syst.* **30** (2017).
13. P. Yger, *et al.*, A spike sorting toolbox for up to thousands of electrodes validated with ground truth recordings in vitro and in vivo. *Elife* **7**, e34518 (2018).
14. M. Saif-ur-Rehman, *et al.*, SpikeDeep-classifier: a deep-learning based fully automatic offline spike sorting algorithm. *J. Neural Eng.* **18**, 16009 (2021).
15. I. Y. Park, *et al.*, Deep learning-based template matching spike classification for extracellular recordings. *Appl. Sci.* **10**, 301 (2019).
16. R. Q. Quiroga, Z. Nadasdy, Y. Ben-Shaul, Unsupervised Spike Detection and Sorting with Wavelets and Superparamagnetic Clustering. *Neural Comput.* **16**, 1661–1687 (2004).
17. L. Huang, *et al.*, WMsorting: wavelet packets' decomposition and mutual information-based spike sorting method. *IEEE Trans. Nanobioscience* **18**, 283–295 (2019).
18. C. R. Caro-Martín, J. M. Delgado-García, A. Gruart, R. Sánchez-Campusano, Spike sorting based on shape, phase, and distribution features, and K-TOPS clustering with validity and error indices. *Sci. Rep.* **8**, 17796 (2018).
19. J. Eom, *et al.*, Deep-learned spike representations and sorting via an ensemble of auto-encoders. *Neural Networks* **134**, 131–142 (2021).
20. Z. Li, Y. Wang, N. Zhang, X. Li, An accurate and robust method for spike sorting based on convolutional neural networks. *Brain Sci.* **10**, 835 (2020).
21. M. Pachitariu, N. Steinmetz, S. Kadir, M. Carandini, K. Harris, Fast and Accurate Spike Sorting of High-Channel Count Probes with KiloSort in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16., (Curran Associates Inc., 2016), pp. 4455–4463.
22. M. Pachitariu, S. Sridhar, J. Pennington, C. Stringer, Spike sorting with Kilosort4. *Nat. Methods* **21**, 914–921 (2024).
23. L. McInnes, J. Healy, N. Saul, L. Großberger, UMAP: Uniform Manifold

Approximation and Projection. *J. Open Source Softw.* **3**, 861 (2018).

24. J. Healy, L. McInnes, Uniform manifold approximation and projection. *Nat. Rev. Methods Prim.* **4**, 82 (2024).
25. R. J. Gardner, *et al.*, Toroidal topology of population activity in grid cells. *Nature* **602**, 123–128 (2022).
26. N. W. Gouwens, *et al.*, Integrated morphoelectric and transcriptomic classification of cortical GABAergic cells. *Cell* **183**, 935–953 (2020).
27. E. Becht, *et al.*, Dimensionality reduction for visualizing single-cell data using UMAP. *Nat. Biotechnol.* **37**, 38–44 (2019).
28. E. K. Lee, *et al.*, Non-linear dimensionality reduction on extracellular waveforms reveals cell type diversity in premotor cortex. *Elife* **10**, e67490 (2021).
29. F. Scala, *et al.*, Phenotypic variation of transcriptomic cell types in mouse motor cortex. *Nature* **598**, 144–150 (2021).
30. M. G. Stokes, 'Activity-silent' working memory in prefrontal cortex: a dynamic coding framework. *Trends Cogn. Sci.* **19**, 394–405 (2015).
31. Z. Friedenberger, E. Harkin, K. Tóth, R. Naud, Silences, spikes and bursts: Three-part knot of the neural code. *J. Physiol.* **601**, 5165–5193 (2023).
32. A. L. Barth, J. F. A. Poulet, Experimental evidence for sparse firing in the neocortex. *Trends Neurosci.* **35**, 345–355 (2012).
33. D. A. Henze, *et al.*, Intracellular Features Predicted by Extracellular Recordings in the Hippocampus In Vivo. *J. Neurophysiol.* **84**, 390–400 (2000).
34. V. de Lafuente, R. Romo, Neural correlate of subjective sensory experience gradually builds up across cortical areas. *PNAS* **103**, 14266–14271 (2006).
35. H. G. Rey, C. Pedreira, R. Quian Quiroga, Past, present and future of spike sorting techniques. *Brain Res. Bull.* **119**, 106–117 (2015).
36. R. J. G. B. Campello, D. Moulavi, J. Sander, Density-Based Clustering Based on Hierarchical Density Estimates BT - Advances in Knowledge Discovery and Data Mining in J. Pei, V. S. Tseng, L. Cao, H. Motoda, G. Xu, Eds. (Springer Berlin Heidelberg, 2013), pp. 160–172.
37. L. McInnes, J. Healy, S. Astels, hdbscan: Hierarchical density based clustering. *J. Open Source Softw.* **2**, 205 (2017).
38. R. Bellman, R. E. Bellman, R. Corporation, *Dynamic Programming* (Princeton University Press, 1957).
39. B. Ghojogh, M. Crowley, F. Karray, A. Ghodsi, *Elements of Dimensionality*

Reduction and Manifold Learning (Springer, 2023).

40. Y. Yang, *et al.*, Dimensionality reduction by UMAP reinforces sample heterogeneity analysis in bulk transcriptomic data. *Cell Rep.* **36**, 109442 (2021).
41. J. Lee, *Introduction to topological manifolds* (Springer Science & Business Media, 2010).
42. E. Schneidman, *et al.*, Synergy from Silence in a Combinatorial Neural Code. *J. Neurosci.* **31**, 15732 LP – 15741 (2011).
43. H. Díaz, *et al.*, Contextual neural dynamics during time perception in the primate ventral premotor cortex. *PNAS* **122**, e2420356122 (2025).
44. A. Zainos, *et al.*, Time Perception VPC Dataset. (2024). Available at: https://doi.org/10.1101/2024.10.03.616577.
45. A. Zainos, *et al.*, Bimodal Detection Task VPC. (2024). Available at: https://doi.org/10.1101/2024.12.11.628069.
46. B. Andrade-Ortega, *et al.*, Multi-Stable Bimodal Perceptual Coding within the Ventral Premotor Cortex. *bioRxiv* 2024.12.11.628069 (2024). https://doi.org/10.1101/2024.12.11.628069.
47. Y. Zhang, D. Han, Y. Wang, Y. Gu, D. Li, SimSort: A Powerful Framework for Spike Sorting by Large-Scale Electrophysiology Simulation. *arXiv Prepr. arXiv2502.03198* (2025).
48. E. M. Trautmann, *et al.*, Accurate Estimation of Neural Population Dynamics without Spike Sorting. *Neuron* **103**, 292–308 (2019).
49. R. Rossi-Pool, R. Romo, Low Dimensionality, High Robustness in Neural Population Dynamics. *Neuron* **103**, 177–179 (2019).
50. P. Papale, F. Wang, M. W. Self, P. R. Roelfsema, An extensive dataset of spiking activity to reveal the syntax of the ventral stream. *Neuron* **113**, 539-553.e5 (2025).
51. D. Kobak, J. L. Pardo-Vazquez, M. Valente, C. K. Machens, A. Renart, State-dependent geometry of population activity in rat auditory cortex. *Elife* **8**, e44526 (2019).
52. K. D. Harris, R. Q. Quiroga, J. Freeman, S. L. Smith, Improving data quality in neuronal population recordings. *Nat. Neurosci.* **19**, 1165–1174 (2016).
53. C. Diekman, K. Dasgupta, V. Nair, K. P. Unnikrishnan, Discovering functional neuronal connectivity from serial patterns in spike train data. *Neural Comput.* **26**, 1263–1297 (2014).

54. C. Baldassano, D. M. Beck, L. Fei-Fei, Parcellating connectivity in spatial maps. *PeerJ* **3**, e784 (2015).
55. C. Baker, E. Froudarakis, D. Yatsenko, A. S. Tolias, R. Rosenbaum, Inference of synaptic connectivity and external variability in neural microcircuits. *J. Comput. Neurosci.* **48**, 123–147 (2020).
56. S. Laquitaine, M. Imbeni, J. Tharayil, J. B. Isbister, M. W. Reimann, Spike sorting biases and information loss in a detailed cortical model. *bioRxiv* 2012–2024 (2024).
57. Y. Zhang, *et al.*, Bypassing spike sorting: Density-based decoding using spike localization from dense multielectrode probes. *Adv. Neural Inf. Process. Syst.* **36**, 77604–77631 (2023).
58. A. Soleymankhani, V. Shalchyan, A new spike sorting algorithm based on continuous wavelet transform and investigating its effect on improving neural decoding accuracy. *Neuroscience* **468**, 139–148 (2021).
59. D. Issar, R. C. Williamson, S. B. Khanna, M. A. Smith, A neural network for online spike classification that improves decoding accuracy. *J. Neurophysiol.* **123**, 1472–1485 (2020).
60. E. Sedaghat-Nejad, *et al.*, P-sort: an open-source software for cerebellar neurophysiology. *J. Neurophysiol.* **126**, 1055–1075 (2021).
61. E. Salinas, A. Hernández, A. Zainos, R. Romo, Periodicity and Firing Rate As Candidate Neural Codes for the Frequency of Vibrotactile Stimuli. *J. Neurosci.* **20**, 5503–5515 (2000).
62. T. M. Cover, *Elements of information theory* (John Wiley & Sons, 1999).
63. M. A. Montemurro, R. Senatore, S. Panzeri, Tight Data-Robust Bounds to Mutual Information Combining Shuffling and Model Selection Techniques. *Neural Comput.* **19**, 2913–2957 (2007).
64. E. Maris, R. Oostenveld, Nonparametric statistical testing of EEG- and MEG-data. *J. Neurosci. Methods* **164**, 177–190 (2007).

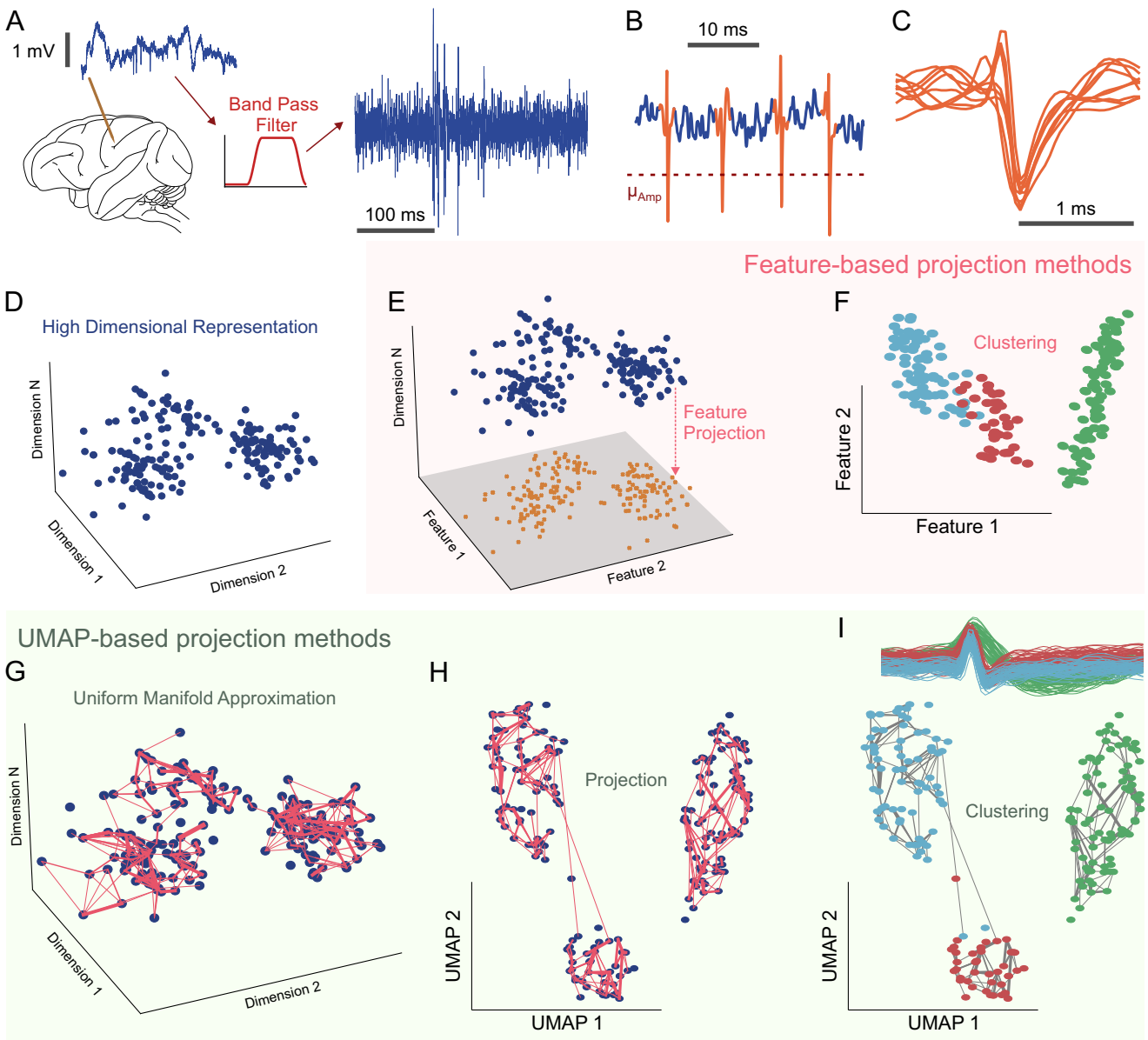

Figure 1. Comparison of feature-based and UMAP-based spike sorting for neuron classification. (A) High-pass filtering (500 Hz–2 kHz) removes low-frequency noise from extracellular signals. (B) Thresholding the pronounced deflections in the filtered data identifies spikes. (C) Windows around each spike, centered on the trough, are isolated to standardize waveform comparisons. (D) Each spike waveform becomes a point in a high-dimensional space (time vs. voltage). (E) Feature-based sorting relies on linear dimensionality reduction methods (e.g., PCA, wavelets), yet can be thrown off by spikes from different neurons. (F) Clustering in this simplified feature space may misclassify neurons due to these linear constraints. (G) UMAP-based sorting, however, employs a nonlinear approach that preserves both local and global structure. (H) This UMAP projection keeps distinct clusters and neuron-specific features more intact. (I) Clustering in the UMAP-projected space better distinguishes individual neurons (shown as distinct colored clusters), reducing merging errors seen with feature-based methods.

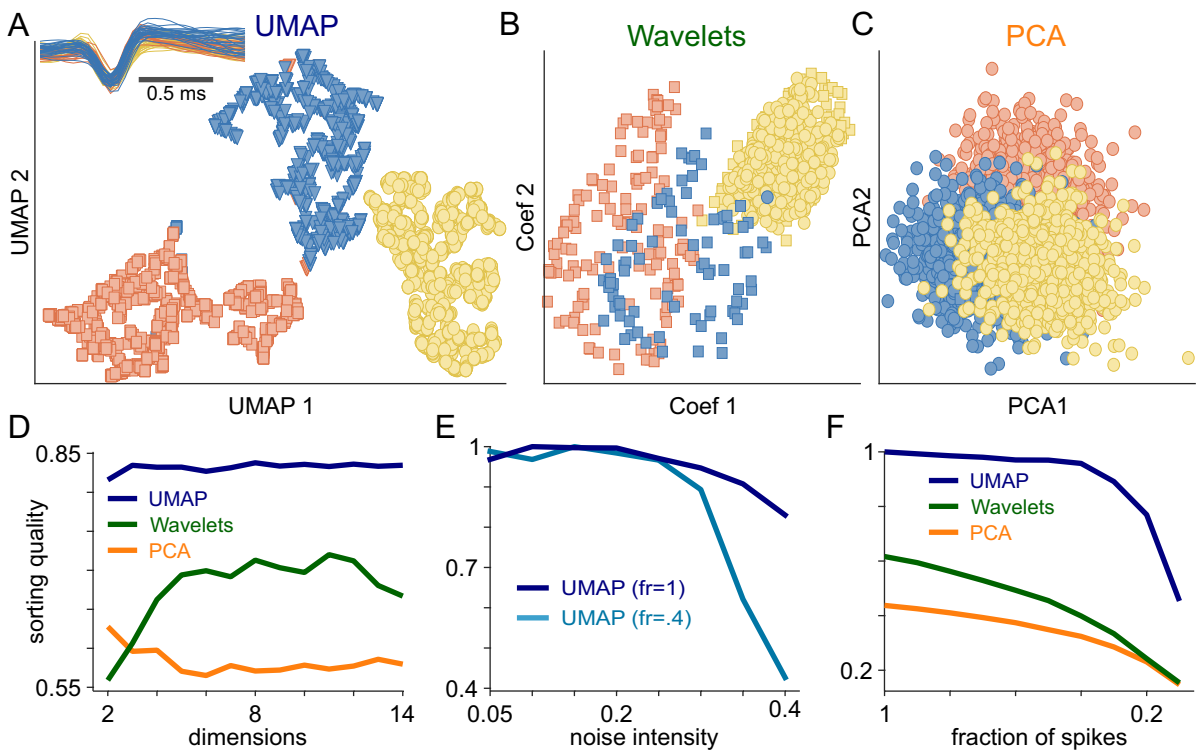

Figure 2. Comparison of UMAP-based and feature-based spike sorting on synthetic data. (A) In a simulated dataset containing three different neurons, UMAP-based sorting forms three well-separated clusters with distinct shapes and colors, indicating strong accuracy and minimal overlap. **(B)** Applying wavelet sorting to the same dataset merges the spikes into just two clusters that exhibit noticeable overlap and weak separation, as illustrated in both 3D (top) and 2D (bottom) views. **(C)** PCA-based sorting isolates only a single cluster with poor separation, underscoring PCA's difficulty with complex spike distributions. **(D)** By contrast, UMAP spike sorting consistently surpasses both wavelet and PCA methods, particularly in higher dimensional spaces. **(E)** UMAP-based sorting also proves robust against noise perturbations at various spike dilution levels (dark vs. light blue traces). Even under increased noise and cluster dilution, it preserves high-quality sorting. **(F)** Moreover, UMAP-based sorting (dark blue) remains accurate despite data loss, whereas wavelet (green) and PCA (orange) methods experience a significant decline in performance.

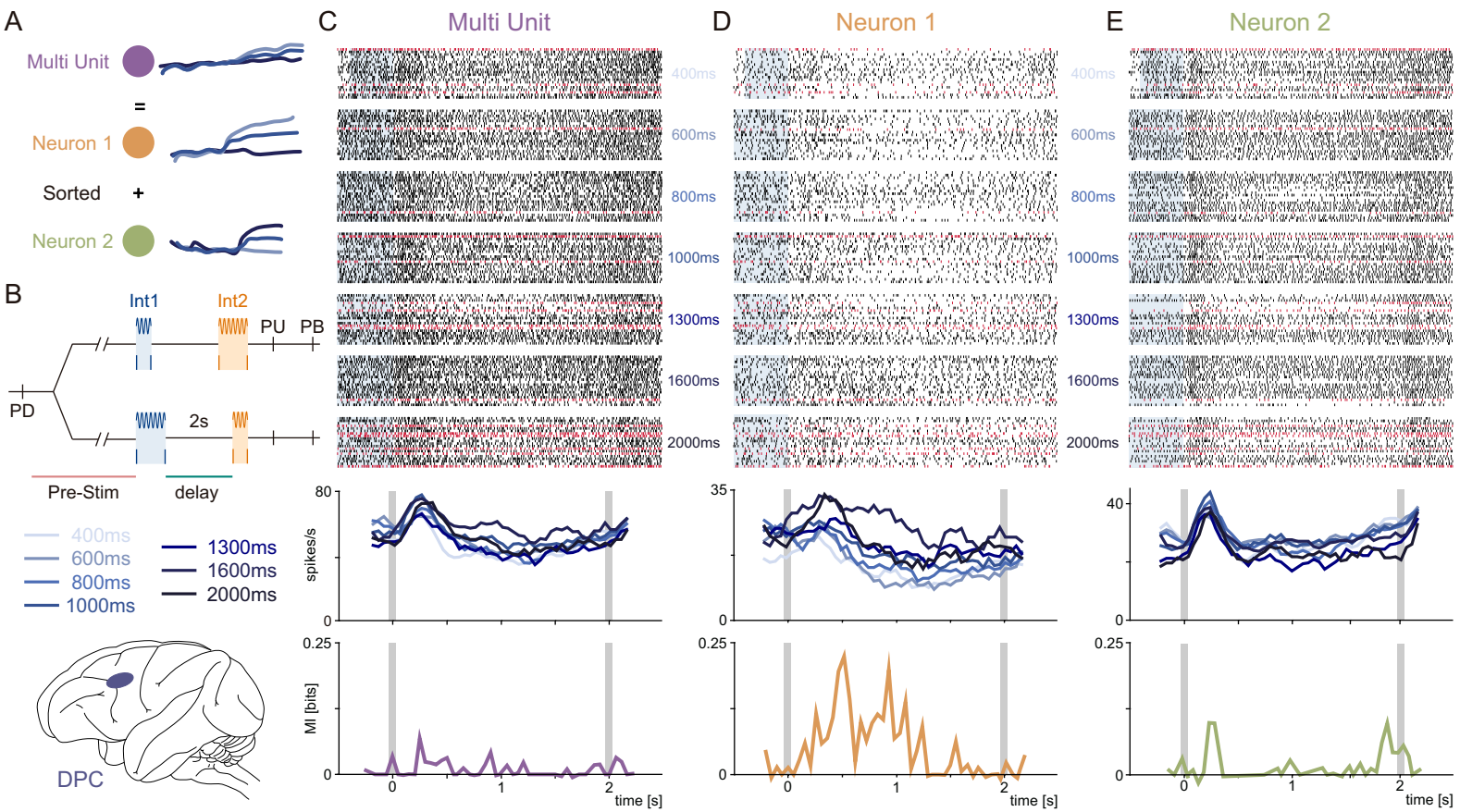

Figure 3. Impact of multi-unit merging on neural information encoding. Multi-unit merging, a common artifact in spike sorting, combines spikes from multiple neurons into a single "putative neuron". This inflates firing rates—particularly masking low firing rate neurons—and obscures their distinct information-encoding properties. (A) Schematic illustrating how merged activity from two true neurons can appear task-irrelevant, as the combined firing pattern shows little difference between two task conditions (green & orange vs purple). Proper sorting reveals each neuron encodes unique information. (B) Time Interval Comparison Task (TICT). Monkeys compare two intervals (Int1 and Int2) separated by a 2-second delay, then press one of two buttons to indicate which interval was longer. Int1 (400 to 2000 ms) is shown in shades of blue. Data was recorded in dorsal prefrontal cortex (DPC, shown in brain cartoon). (C) Top: Raster plot of a misclassified multi-unit in the DPC, aligned to the end of (t=0 = start of the delay). Gray rectangles indicate Int1 backpropagated from t=0. Bottom: The average firing rate and mutual information fail to show meaningful encoding. (D) The first neuron contributing to the multi-unit (panel C). Despite a low firing rate, it exhibits a clear positive encoding of Int1 (larger stimulus intensities produce higher firing rates). (E) The second neuron shows a strong early sensory response followed by a negative encoding of Int1. Merging these two distinct patterns masks each neuron's specific role in information processing.

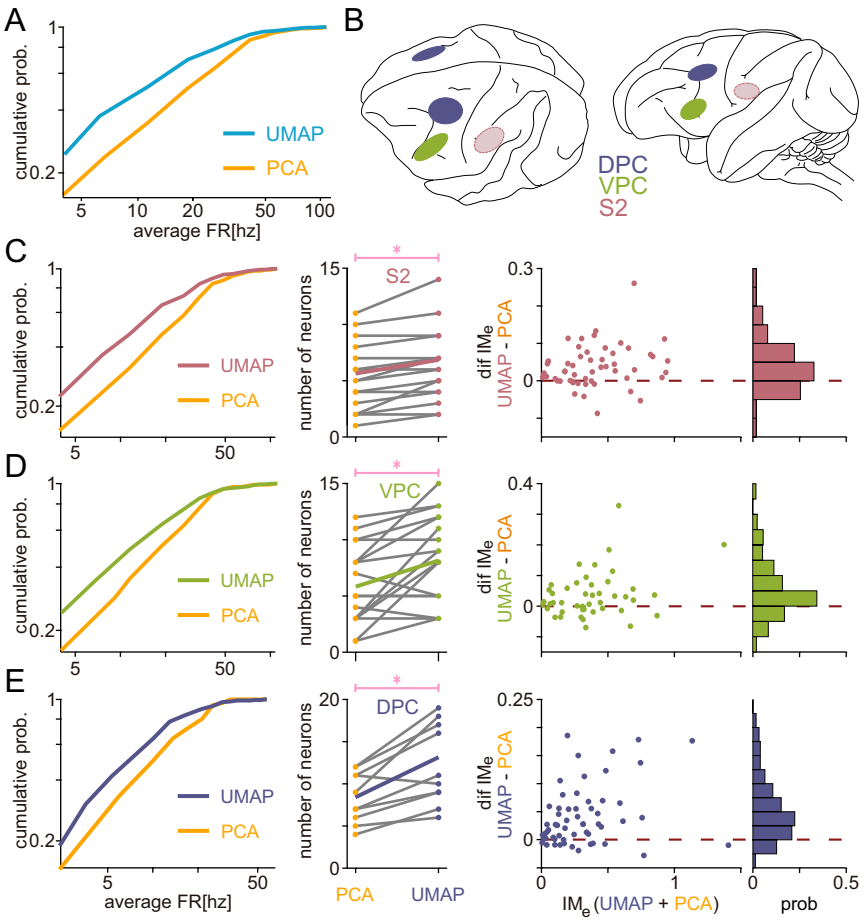

Figure 4. UMAP-based sorting enhances detection of low-firing rate neurons and boosts mutual information. (A) Cumulative average firing rate distributions for UMAP-based and PCA-based spike sorting reveal that UMAP identifies more low-firing neurons, reflecting greater sensitivity to sparse firing. (B) Anatomical maps illustrate three cortical regions: ventral premotor cortex (VPC, green), secondary somatosensory cortex (S2, red), and dorsal prefrontal cortex (DPC, blue). Dotted lines in the lightly shaded oval above S2 indicate recording sites in deeper cortical layers—areas involved in higher-level processing and decision-making. (C–E) Comparisons of PCA and UMAP sorting across these regions. Left: UMAP consistently recovers a larger number of low-firing neurons based on cumulative firing rate distributions. **Center:** It also yields higher overall neuron counts from the same sessions. Right: UMAP reveals greater mutual information (Shannon's MI) compared to PCA, indicating stronger information decoding across all brain regions. In summary, UMAP effectively increases neuron recovery, captures low-firing activity, and enhances the information gained from cognitive task data.

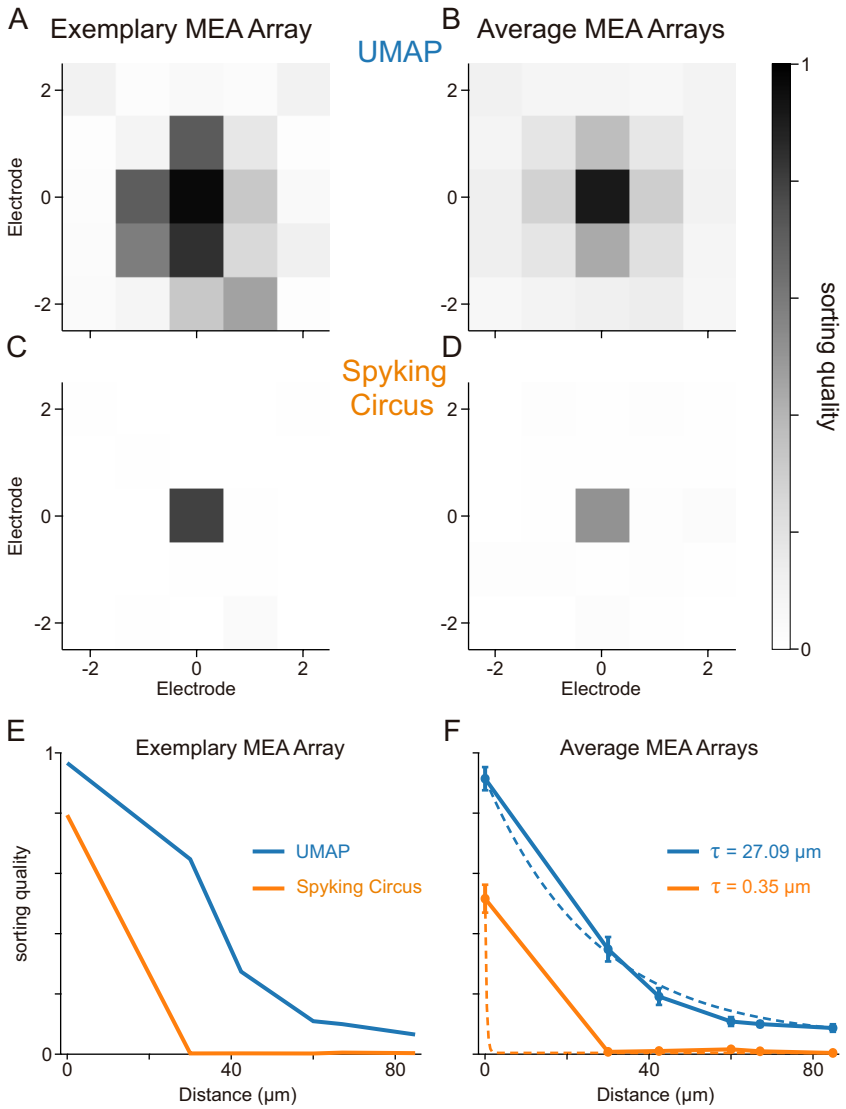

Figure 5. Spatiotemporal spike sorting on MEA data: UMAP vs. SpiKING CIRCUS. We compare spatial sorting quality on multi-electrode array (MEA) data using UMAP-based sorting vs. SpiKING CIRCUS, with intracellular recordings as ground truth (GT). Sorting quality metric ($Q_{i,GT}$) approximates the degree to which each electrode can detect GT spikes (see Fig. S2). (A) Spatial matrix of $Q_{i,GT}$ for one MEA recording using UMAP-based sorting. The gradual decline in values with the distance from the electrode indicates that spike detection is still reliable at larger separations. (B) The average $Q_{i,GT}$ matrix from different MEA recordings shows a similar gradual decrease, emphasizing UMAP's ability to detect signals at larger distances from the GT neuron. (C, D) In contrast, SpiKING CIRCUS detects GT spikes successfully only on the electrodes directly above the neuron, whereas neighboring electrodes show drastic drops in $Q_{i,GT}$, indicating a limited spatial influence. (E) UMAP maintains better detection quality with distance, while SpiKING CIRCUS breaks down at closer distances. (F) Across all MEAs, SpiKING CIRCUS's decay rate is nearly two orders of magnitude higher, showing that UMAP-based sorting recovers GT spikes consistently across tens of micrometers, while the spatial accuracy of SpiKING CIRCUS degrades quickly.

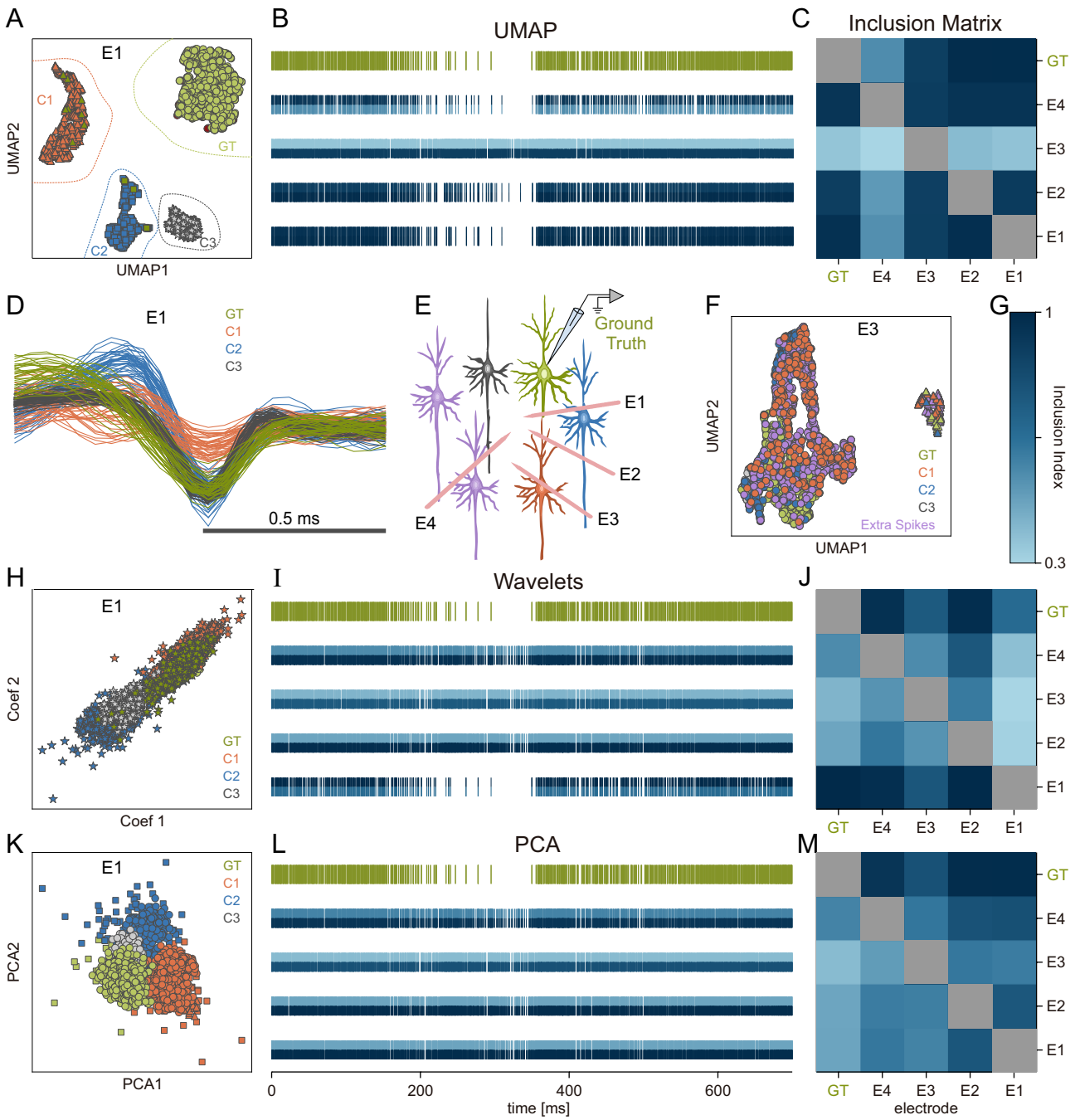
A
E1
C1
GT
C2
C3
UMAP2
UMAP1
B
UMAP
C
Inclusion Matrix
GT
E4
E3
E2
E1
D
E1
GT
C1
C2
C3
0.5 ms
E
Ground Truth
E1
E2
E3
E4
F
E3
UMAP2
UMAP1
GT
C1
C2
C3
Extra Spikes
G
1
0.3
Inclusion Index
H
E1
Coef 2
Coef 1
GT
C1
C2
C3
I
Wavelets
J
K
E1
PCA2
PCA1
GT
C1
C2
C3
L
PCA
0
200
400
600
time [ms]
M
electrode

Figure 6. Comparative analysis of spike sorting using UMAP vs. feature-based methods in *In Vivo* extracellular recordings. Recordings from Buzsaki's lab (26) include an intracellular electrode serving as ground truth (GT, green) and four extracellular electrodes (E1–E4) placed at different distances from the GT neuron. (A) UMAP-based sorting on electrode E1 identifies four clusters (distinct shapes). One cluster (S_{E1}, circular markers) overlaps substantially with the GT neuron, while the other three represent separate putative neurons. (B) Raster plots for GT clusters found by UMAP on each extracellular electrode, alongside the GT neuron's raster. The top color of each line represents the percentage of cluster spikes that come from the GT ($I_{E,GT}$), and the bottom color indicates the percentage of GT spikes captured by that cluster ($I_{GT,E}$). Electrodes E1 and E2 align well with the GT neuron, whereas E3's cluster is contaminated, and E4's cluster loses many GT spikes. (C) Inclusion Index matrix $M_{i,j}$ for UMAP, illustrating overlap of GT spikes across electrode clusters. Darker squares for E1 and E2 reflect accurate capture of GT spikes, while E3 and E4 show contamination and loss, respectively. (D) UMAP-sorted waveforms from E1, color-coded to match the clusters in (A). (E) Experimental schematic from Buzsaki's lab, showing the intracellular GT neuron and four extracellular electrodes. (F) UMAP sorting on electrode E3 highlights significant contamination, with many not-GT spikes mislabeled as GT (shown in a separate color). (G) Color bar for the inclusion matrices (C, J, M). (H-J) Wavelet-based sorting: (H) projection of spikes from E1 in Wavelet feature space, identifying multiple clusters. (I) Raster plots for the GT clusters on each electrode. The GT neuron's pattern is not well isolated. (J) Inclusion matrix shows low, inconsistent overlap for GT spikes across electrodes. (K-M) PCA-based sorting: (K) PCA projection for E1, resulting in broad, overlapping clusters. (L) Raster plots for these GT clusters showing substantial contamination. (M) Inclusion matrix reflects the poor specificity of PCA-based clustering, with larger fractions of non-GT in each putative cluster.

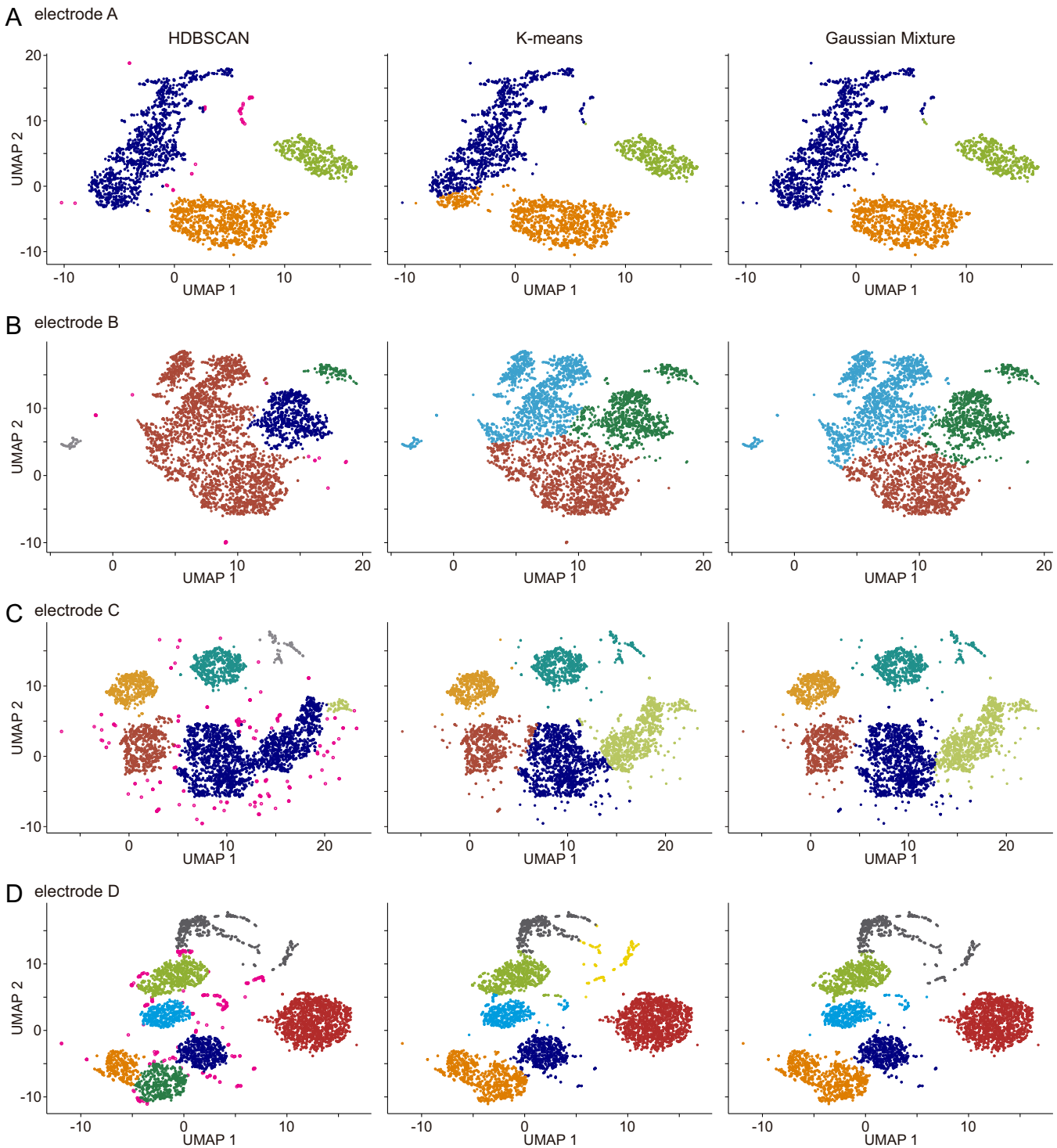

Figure S1. Comparison of three clustering methods—HDBSCAN, K-means, and Gaussian Mixture—for neuronal spike classification. This panel shows how each approach performs on four electrodes (A–D) during a temporal interval comparison task, where different colors denote the spikes assigned to putative neurons. HDBSCAN stands out for two key reasons: (1) it can reveal clusters with unusual shapes and densities (see electrodes A–D), and (2) it avoids forcing uncertain spikes (shown in pink) into any cluster. These unassigned points typically represent noise or non-neuronal signals, and excluding them sharpens the overall accuracy. In contrast, both K-means and Gaussian Mixture assign every spike to a cluster, which often misclassifies ambiguous events. Consequently, HDBSCAN more faithfully represents actual neuronal activity by filtering out only those questionable spikes that impair clarity, surpassing conventional feature-based clustering methods.

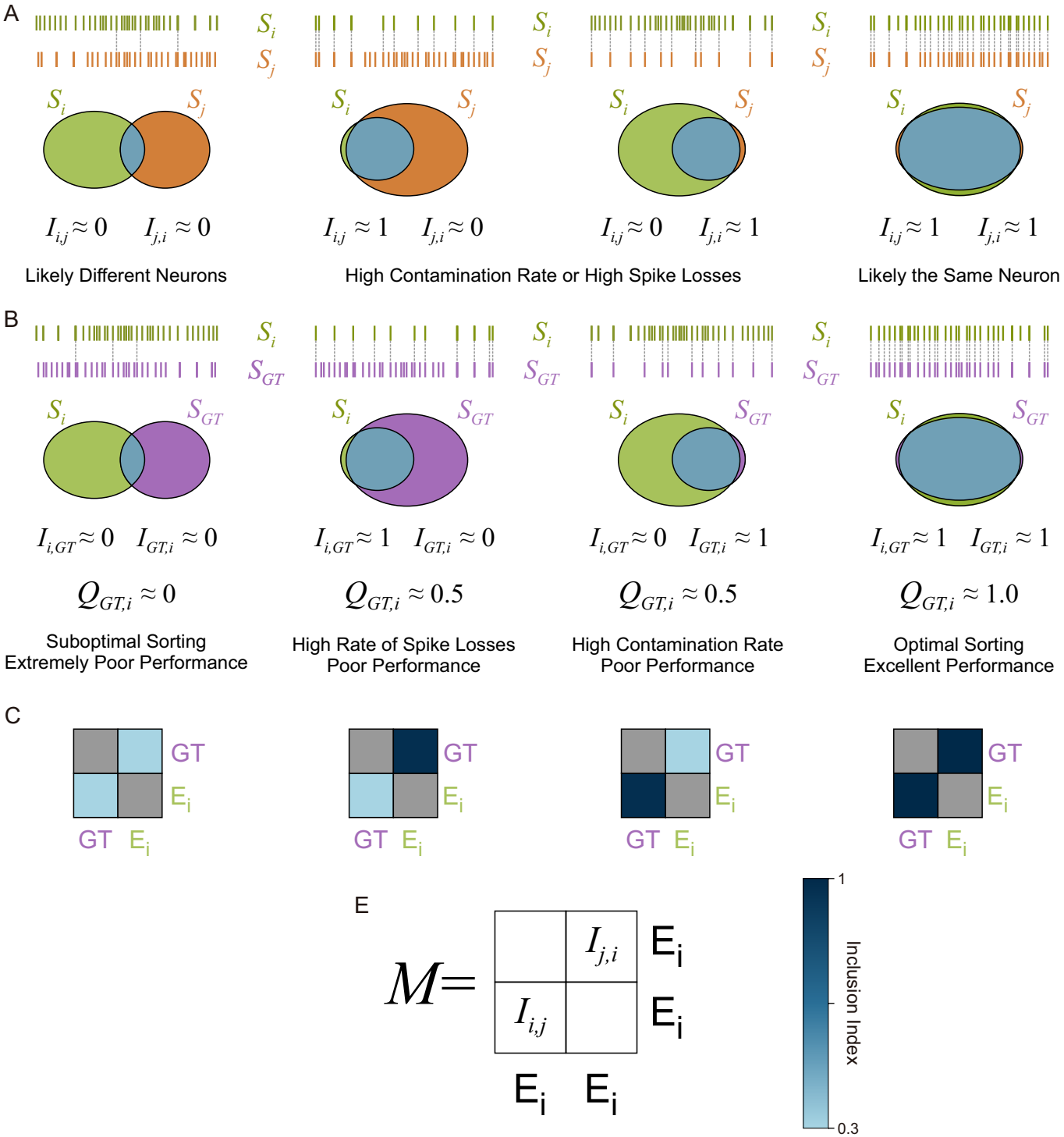

Figure S2. Probabilistic schema of spike separation outcomes. This figure illustrates the way that different spike-separation cases affect the inclusion matrix ($IM_{i,j}$) between an electrode cluster (E1) and a ground truth (GT) neuron. Notice that $IM_{i,j}$ is not symmetric since prob(E1|GT) ≠ prob(GT|E1). Matrix element (2,1) is the fraction of E1 cluster spikes that belong to GT, and (1,2) is the fraction of GT spikes covered by E1. Each panel displays a clear result: (A) Worst Case: Few E1 spikes are GT in origin, and there are numerous missing GT spikes in E1 with hardly any overlap either way. (B) High Contamination: The majority of the GT spikes are found in E1; however, many non-GT spikes are within E1, thereby decreasing prob(E1|GT). (C) High Spike Loss: E1 comprises primarily GT spikes but fails to capture much of this class, thereby decreasing prob(GT|E1). (D) Ideal Situation: Nearly all GT spikes are in E1 (no losses) and the majority of spikes in E1 are from GT (no contamination). (E) Color Bar: Applied to $IM_{i,j}$ values (ass in Figure 3), which are the number of spikes in common between GT and E1. These situations together define how spike contamination or loss would dramatically affect the accuracy of sorting, therefore clearly showing the need for accurate spike assignment.

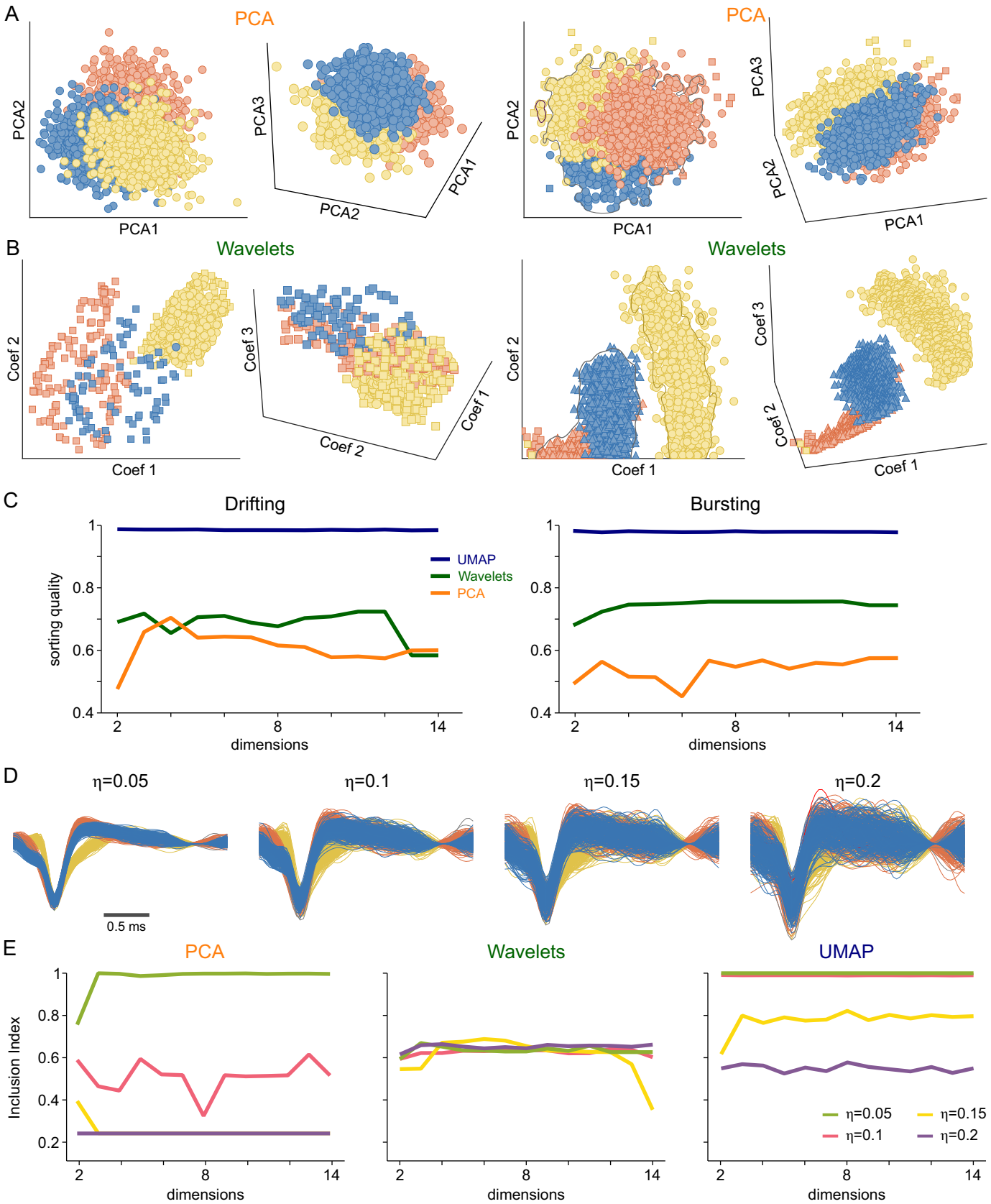
A
PCA
PCA2
PCA1
PCA3
PCA2
PCA1
PCA
PCA2
PCA1
PCA3
PCA2
PCA1
B
Wavelets
Coef 2
Coef 1
Coef 3
Coef 2
Coef 1
Wavelets
Coef 2
Coef 1
Coef 3
Coef 2
Coef 1
C
Drifting
Bursting
sorting quality
1
0.8
0.6
0.4
2
8
14
dimensions
UMAP
Wavelets
PCA
D
η=0.05
η=0.1
η=0.15
η=0.2
0.5 ms
E
PCA
Wavelets
UMAP
Inclusion Index
1
0.8
0.6
0.4
0.2
2
8
14
dimensions
η=0.05
η=0.1
η=0.15
η=0.2

Figure S3. Impact of noise, drifting, and low firing rates on Spike Sorting performance. This figure illustrates how PCA-, Wavelet -, and UMAP-based methods respond to challenges frequently encountered in spike sorting: overlapping waveforms, background noise, drifting, bursting, and neurons with very low firing rates. All clustering was performed with HDBSCAN. (A, B) PCA (A) and Wavelet (B) projections for an example dataset from Quiroga et al. (16). Colors indicate ground truth (GT) neuron identities, while each marker denotes a detected spike. Although adding higher dimensions can sometimes improve cluster separability, it also complicates visualization, and neither PCA nor Wavelet consistently isolates the three GT neurons. (C) Sorting performance under drifting (left) and bursting (right) conditions. UMAP maintains robust performance (blue trace), whereas PCA (orange trace) and Wavelet (green trace) degrade substantially when waveforms shift over time (drifting) or when neurons exhibit bursts of spikes. (D) Sample spike waveforms with added noise levels (η=0.05, 0.1, 0.15, 0.2) to synthetic data from Quian Quiroga et al. (16). UMAP-based sorting is robust to increasing background noise. (E) Sorting performance as a function of projection dimensionality at different noise levels (left/middle/right panels for PCA, Wavelet, and UMAP, respectively). At high noise (η=0.2), UMAP still yields more than 70% accuracy and outperforms both PCA and Wavelet. The same panels also show how each method copes when an increasing fraction of spikes is removed to simulate a “silent” neuron with low firing rates. UMAP maintains high accuracy even when only 20% of that neuron's spikes remain, whereas PCA and Wavelet degrade rapidly.

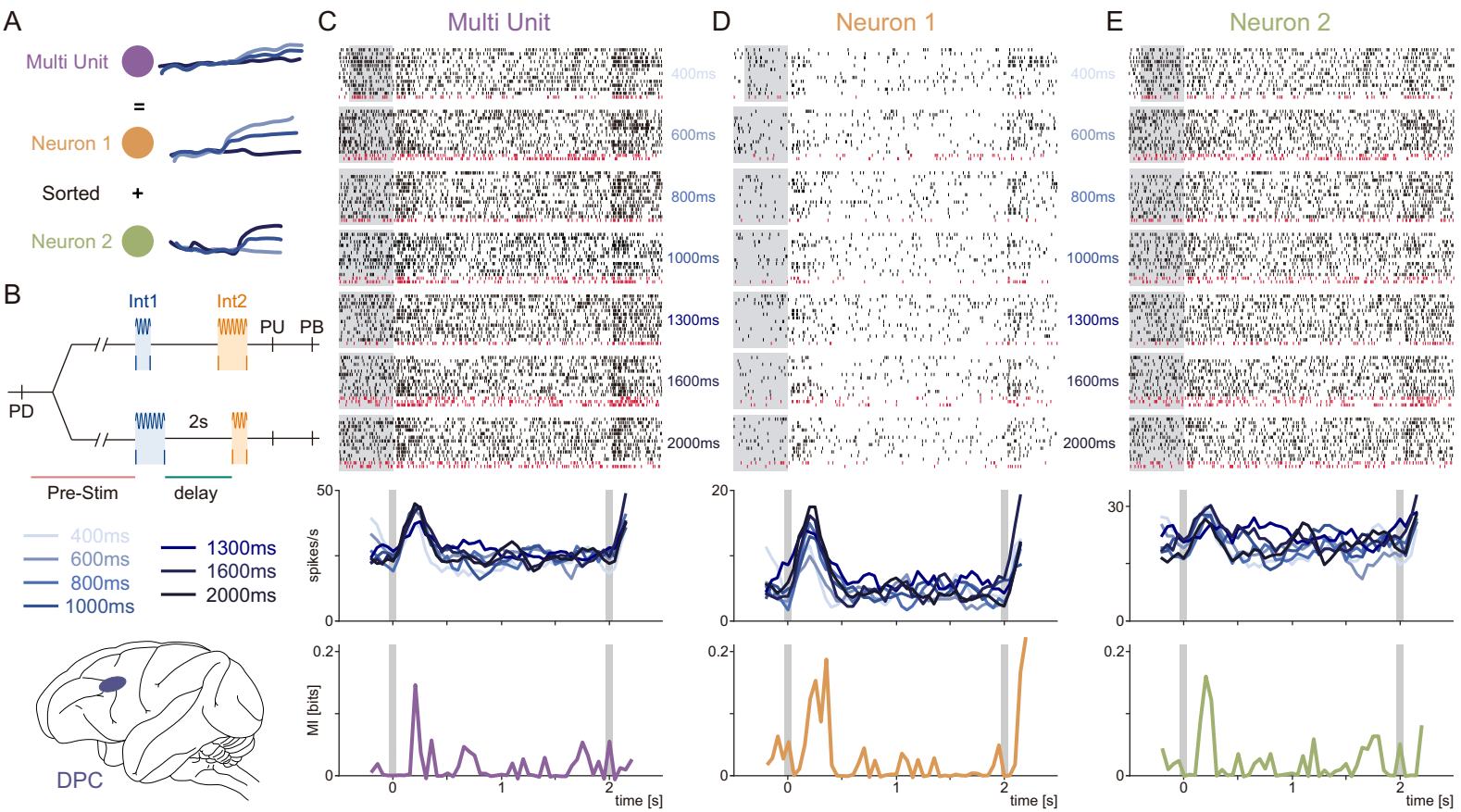

Figure S4. Preserving low-firing-rate neurons with UMAP sorting in the Time Interval Comparison Task. The figure shows how pooling spikes from several neurons into a single muti-unit can eliminate important task-related encoding—particularly if there is a low firing rate neuron. (A) Cartoon showing how poor spike sorting artificially inflates firing rates by combining activity from different neurons, potentially obscuring their distinct encoding patterns. (B) Time Interval Comparison Task (TICT). Animals are comparing two stimulus intervals (Int1 and Int2), both 400–2000 ms, with 2-second gap. The animal must retain Int1 information through the gap and compare it with Int2. Recordings from dorsal prefrontal cortex (DPC). (C) Raster (top) and firing rate (middle) of a multi-unit artificially created by pooling two neurons when using PCA-based sorting. Gray shows the stimulus period, correct responses in black, and errors in red. The mutual information trace (bottom) shows reduced encoding of Int1 since combined units lose valuable signals from low-firing units. (D, E) Single neurons sorted using UMAP that were conflated in (C). While the neuron in (D) fires rarely, however, it does manage to encode Int1 at the start of the delay phase—information lost in the multi-unit representation (C). Conversely, the second neuron (E) has a diverging firing pattern and encodes unique dynamics of the task. These two neurons collectively convey far more information about Int1 than the conflated multi-unit in (C).

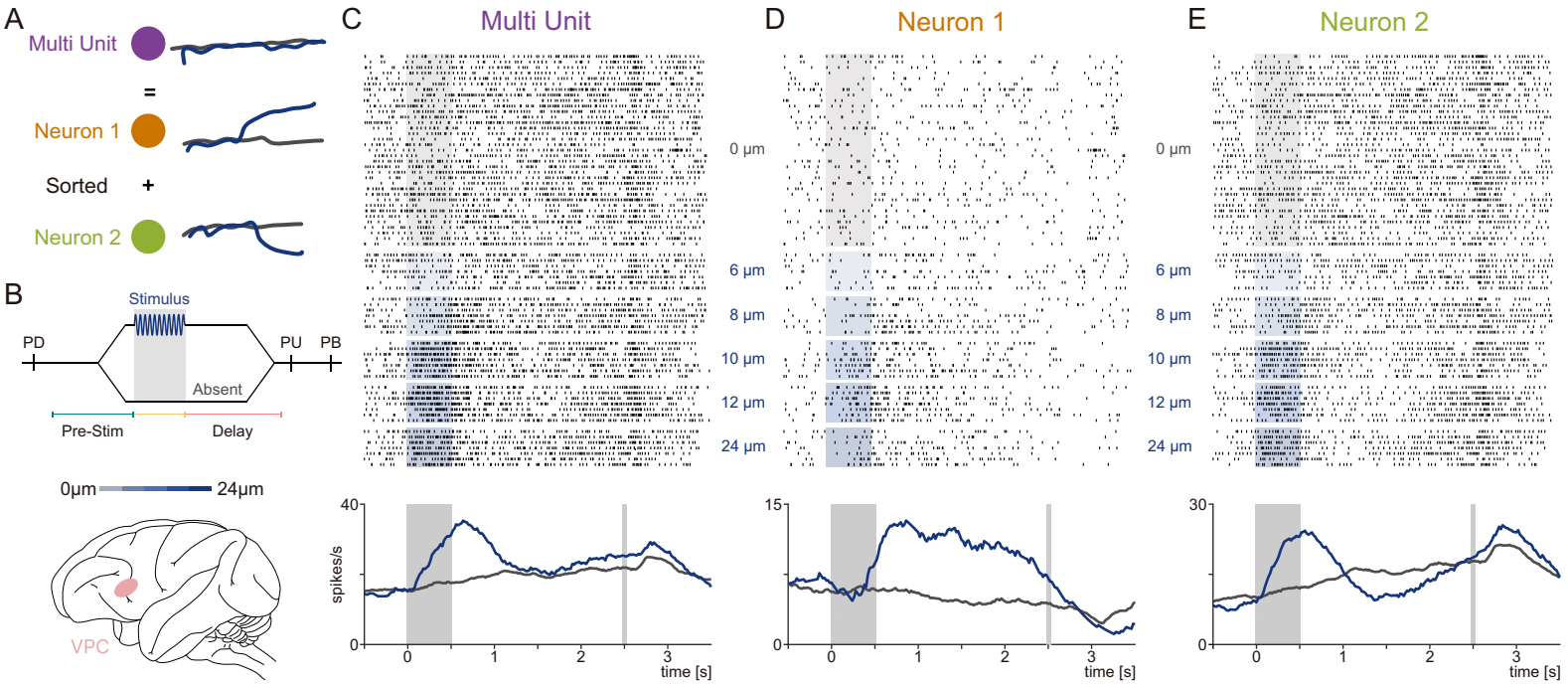

Figure S5. Preserving different encoding patterns with UMAP-based sorting in tactile detection task. (A) Schematic showing how inadequate spike sorting inflates firing rates by combining separate neuronal activities, potentially masking each neuron's invidual encoding. (B) Tactile detection task: Animals receive a vibrotactile stimulus (0–24 µm) on the fingertip and must indicate whether it was present or absent. Recordings are from the ventral premotor cortex (VPC, pink). (C) Raster plots (top) and firing rate (bottom) for multi-unit formed by merging two neurons under PCA-based sorting. (C) Raster plots (top) and firing rate (bottom) for a multi-unit formed by merging two neurons under PCA-based sorting. Trials are aligned to stimulus onset. Blue shading marks stimulus-present trials; gray shading marks stimulus-absent trials. Combining these spikes conceals each neuron's separate response profile, yielding an incomplete picture of their activities. (D, E) UMAP-sorted versions of the two underlying neurons from (C). The neuron in (D) maintains a prolonged response during the delay, while the neuron in (E) responds strongly but briefly. By separating them, UMAP-based sorting retains each neuron's individual activity, avoiding the loss of valuable low-firing-rate signals.

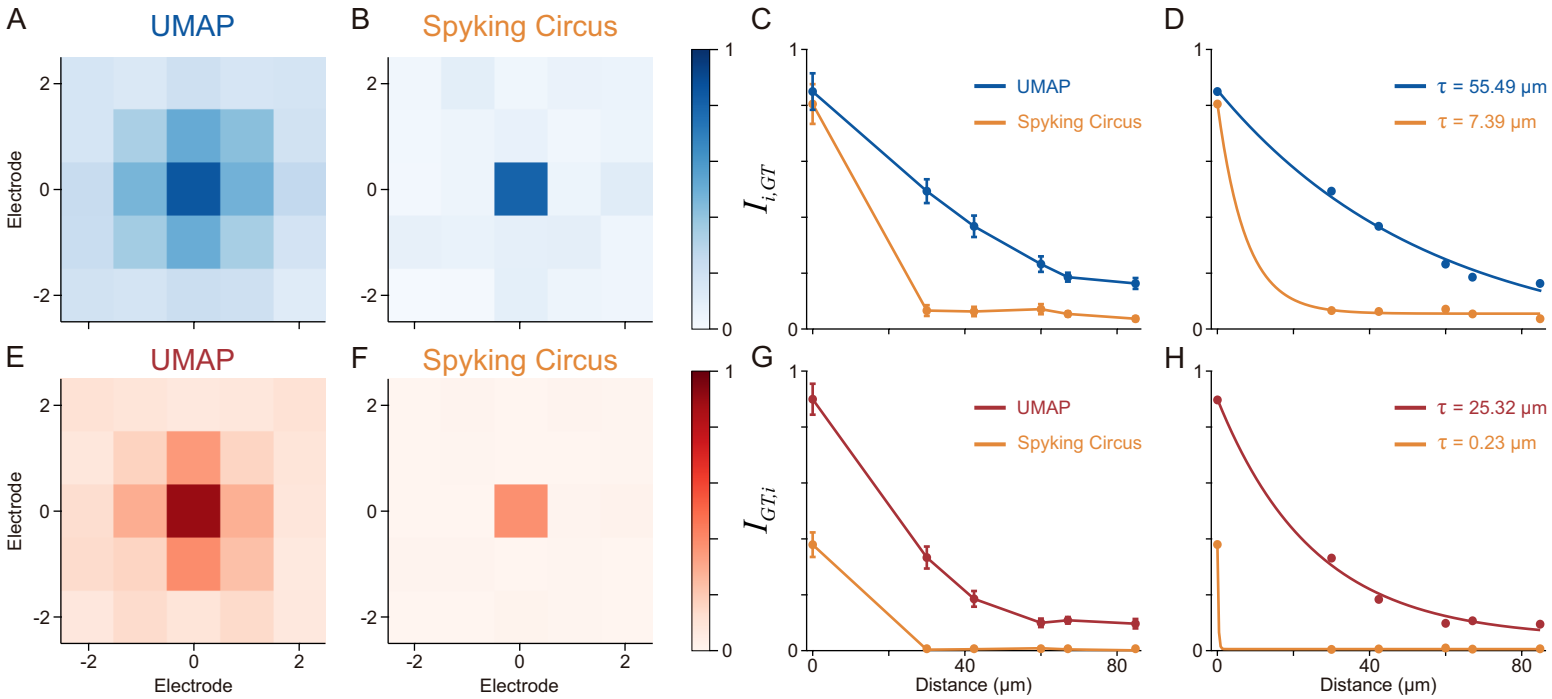

Figure S6. Influence of contamination and lost spikes on MEA sorting accuracy. This figure examines how contamination (misclassify spikes) and lost spikes contribute to the spatial decay of sorting accuracy (Fig. 5) when comparing UMAP-based sorting and against SpiKING CIRCUS on multielectrode array (MEA) recordings. Each panel shows results averaged over multiple MEAs, with an intracellular ground truth (GT) neuron serving as a reference. (A, E) Contamination (A) and lost-spike (E) matrices for UMAP. (B, F) Corresponding matrices for SpyKING CIRCUS. (C, G) Decay of contamination probability ($I_{i,GT}$, the fraction of cluster spikes that belong to GT) and recall probability ($I_{GT,i}$, the fraction of spikes in the cluster) as electrode distance increases. (D, H) Fitted decay rates reveal that SpyKING CIRCUS performance drops rapidly, especially for recall ($I_{GT,i}$), indicating it misses many GT spikes beyond the central electrode. By contrast, UMAP sustains higher accuracy across a wider spatial range, although lost spikes still grow with distance.