

## Finding the Event Structure of Neuronal Spike Trains

**J. Vincent Toups**

*vincent.toups@gmail.com*

*Computational Neurophysics Laboratory, Department of Physics and Astronomy,  
University of North Carolina, Chapel Hill, NC 27599-3255, U.S.A.*

**Jean-Marc Fellous**

*fellous@email.arizona.edu*

*Psychology Department, University of Arizona, Tucson, AZ 85721, U.S.A.*

**Peter J. Thomas**

*pjthomas@case.edu*

*Computational Biomathematics Laboratory, Departments of Mathematics, Biology and  
Cognitive Science, Case Western Reserve University, Cleveland, OH 44106, U.S.A.,  
and Department of Neuroscience, Oberlin College, Oberlin, OH 44074, U.S.A.*

**Terrence J. Sejnowski**

*terry@salk.edu*

*Howard Hughes Medical Institute, The Salk Institute, La Jolla, CA 92186, U.S.A.,  
and Division of Biological Sciences, University of California San Diego,  
La Jolla, CA 92093, U.S.A.*

**Paul H. Tiesinga**

*P.Tiesinga@science.ru.nl*

*Computational Neurophysics Laboratory, Department of Physics and Astronomy,  
University of North Carolina, Chapel Hill, NC 27599, U.S.A., and  
Donders Institute for Brain, Cognition and Behaviour, Radboud University  
Nijmegen, Nijmegen 6500 GL, The Netherlands*

Neurons in sensory systems convey information about physical stimuli in their spike trains. *In vitro*, single neurons respond precisely and reliably to the repeated injection of the same fluctuating current, producing regions of elevated firing rate, termed events. Analysis of these spike trains reveals that multiple distinct spike patterns can be identified as trial-to-trial correlations between spike times (Fellous, Tiesinga, Thomas, & Sejnowski, 2004). Finding events in data with realistic spiking statistics is challenging because events belonging to different spike patterns may overlap. We propose a method for finding spiking events that uses contextual information to disambiguate which pattern a trial belongs to. The procedure can be applied to spike trains of the same neuron across

**multiple trials to detect and separate responses obtained during different brain states. The procedure can also be applied to spike trains from multiple simultaneously recorded neurons in order to identify volleys of near-synchronous activity or to distinguish between excitatory and inhibitory neurons. The procedure was tested using artificial data as well as recordings in vitro in response to fluctuating current waveforms.**

## 1 Introduction

---

Sensory information is encoded in the spatiotemporal pattern of spiking activity in populations of neurons (Ermentrout, Galan, & Urban, 2008; Tiesinga, Fellous, & Sejnowski, 2008). A commonly used measure for neural activity in a single neuron is the firing rate (Hubel & Wiesel, 1962): the number of spikes per second that a neuron produces. Under some circumstances, however, cortical neurons produce precise spike times across repeated stimulation by the same stimulus (Bair & Koch, 1996; Buracas, Zador, DeWeese, & Albright, 1998; Haider & McCormick, 2009; Reinagel, Godwin, Sherman, & Koch, 1999; Reinagel & Reid, 2000, 2002). We will refer to the resulting temporally localized concentrations of spike time density across trials as spike time events or simply events. Such events presumably encode some aspects of the stimulus because they are produced reliably from trial to trial, and their exact timing depends on the specific stimulus.

A representative example rastergram and histogram obtained from a model are shown in Figures 1A and 1B. The rastergram is characterized by vertical bands of aligned spike times. Spikes within these bands occur on almost every trial at approximately the same time with respect to stimulus onset; each band corresponds to a peak in the histogram. Each such line or peak is a spike time event. We characterize these events using measures that quantify the likelihood of occurrence of a spike during an event (reliability) and the precision of this spike when it occurs (precision).

Algorithms have been proposed to identify events (Mainen & Sejnowski, 1995; Tiesinga, Fellous, & Sejnowski, 2002), but they often fail to separate overlapping events (see Figure 1B, double-headed arrow). Overlapping events occur because of within-trial correlations in the form of spike time patterns. These patterns arise due to the refractory period and afterhyperpolarization currents. We hypothesize that the separation of overlapping events can be improved by taking into account contextual information contained in the spiking history, which captures local within-trial correlations. Figure 1C shows an example of such within-trial correlations aiding the separation of overlapping events. Although the events were overlapping when the histogram was calculated across all trials (see Figure 1B, double-headed arrow), they were clearly nonoverlapping when trials were reordered according to the spike pattern they belonged to (see Figure 1C). Using the contextual information from neighboring spike times makes the procedure

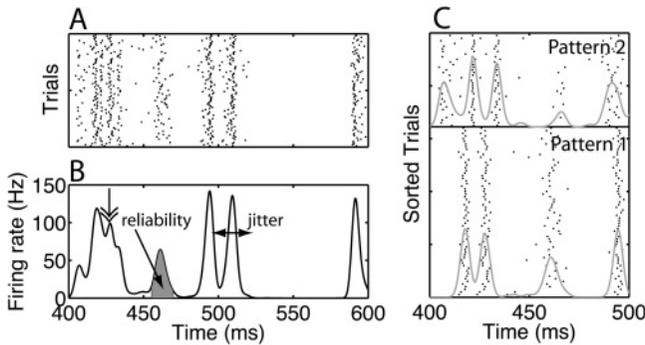


Figure 1: Spike trains can be characterized in terms of events. (A) A rastergram with spike trains obtained from an example model neuron in response to the same fluctuating current waveform presented multiple times (Tiesinga & Troups, 2005). The spike trains vary from trial to trial because of an independent noise current (Tiesinga & Troups, 2005). In a rastergram, each spike is represented by a tick, with its  $x$ -ordinate being the spike time and the  $y$ -ordinate being the trial number. In the rastergram, vertical bands correspond to spike alignments, each of which is an event. (B) The spike time histogram is the number of spikes that fall in a bin, normalized by the number of trials and the bin width so that the result is a firing rate expressed in Hz. Spike alignments (events) lead to peaks in the histogram. The spike time jitter of an event is the width of the corresponding peak, which can be estimated as the standard deviation of the spike times in the event. Sometimes we will use precision instead, which is numerically equal to the inverse of the jitter. The area under the peak is proportional to the reliability, the fraction of trials on which there was a spike during the event. When peaks are isolated, it is easy to assign each spike to an event. However, if there are overlapping peaks, such as the peak indicated by the double arrow, event assignment is much more difficult. In this article, we introduce a technique, based on detecting spike patterns, which can efficiently separate events. (C) When the same spike trains as in panel A are sorted using the clustering algorithm (see section 3.3), spike patterns emerge. The horizontal line separates two spike patterns, and the gray curves are the corresponding histograms. The aggregate histogram in panel B is a weighted sum of the two histograms shown separately in panel C.

more powerful, but also more computationally expensive because it involves additional steps: spike patterns have to be detected and events common to multiple patterns have to be merged. In this article, we introduce heuristics for the event-based analysis that reduce the computational load and consequently allow semiautomatic analysis of large volumes of spike train data.

When applied to an ensemble of spike trains from repeated trials, our method characterizes events by their time of occurrence, their precision,

and their reliability. In the following, we will use *precision* and *jitter* interchangeably to refer to the temporal resolution of spike times. The precision (units: 1/ms) is defined to be the inverse of the jitter (units: ms). An advantage of the event-based analysis is that it does not rely on fitting a specific parametric model for neural dynamics (Jolivet, Lewis, & Gerstner, 2004; Jolivet, Rauch, Luscher, & Gerstner, 2006; Keat, Reinagel, Reid, & Meister, 2001; Pillow, Paninski, Uzzell, Simoncelli, & Chichilnisky, 2005); rather it models the data directly.

The method was applied to data from layer 5 cortical neurons recorded in vitro. Neurons were driven by a fixed fluctuating current waveform with an amplitude that was varied systematically from trial to trial. The neurons locked to the current drive, which means that they produced spiking events at specific times relative to stimulus onset. We found that the spike trains changed with amplitude in a way that preserved information about the stimulus time course. With high likelihood, each trial fell into one of a small selection of distinct patterns. The frequency with which a given cell exhibited each pattern varied systematically as a function of stimulus amplitude. Hence, the information about the amplitude was implicitly encoded in the trial-to-trial variability.

The article is organized as follows. Section 2 reviews our general experimental procedures, stimulus generation, and experimental design. Section 3 describes general elements of the analysis procedure, including the fuzzy clustering algorithm used, the calculation of the Victor-Purpura distance, and calculation of the entropy and mutual information between classifications of spike train ensembles. Section 4 begins with an overview of the method as a whole and then presents each component in detail in section 4.1, as well as a detailed application of the method to experimental data in section 4.2. We conclude with a general discussion in section 5.

## 2 Experimental Procedures

---

**2.1 General Experimental Procedures.** The voltage responses of cortical neurons were measured in a rat slice preparation as described previously (Fellous et al., 2001). The Salk Institute Animal Care and Use Committee approved protocols for these experiments; the procedures conform to USDA regulations and NIH guidelines for humane care and use of laboratory animals. Briefly, coronal slices of rat prelimbic and infralimbic areas of prefrontal cortex were obtained from 2- to 4-week-old Sprague-Dawley rats. Rats were anesthetized with isoflurane and decapitated. Their brains were removed and cut into 350  $\mu\text{m}$  thick slices on a Vibratome 1000 (EB Sciences, Agawam, MA). Slices were then transferred to a submerged chamber containing standard artificial cerebrospinal fluid (ACSF, mM: NaCl, 125;  $\text{NaH}_2\text{CO}_3$ , 25; D-glucose, 10; KCl, 2.5;  $\text{CaCl}_2$ , 2;  $\text{MgCl}_2$ , 1.3;  $\text{NaH}_2\text{PO}_4$ , 1.25) saturated with 95%  $\text{O}_2$ /5%  $\text{CO}_2$ , at room temperature. Whole cell patch clamp recordings were achieved using glass electrodes (4–10  $\text{M}\Omega$ ;

containing, in mM: KmeSO<sub>4</sub>, 140; Hepes, 10; NaCl, 4; EGTA, 0.1; Mg-ATP, 4; Mg-GTP, 0.3; Phosphocreatine, 14). Patch-clamp was performed under visual control at 30 to 32°C. In most experiments, Lucifer Yellow (RBI, 0.4%) or Biocytin (Sigma, 0.5%) was added to the internal solution for morphological identification. In all experiments, synaptic transmission was blocked by D-2-amino-5-phosphonovaleric acid (D-APV; 50 μM), 6,7-dinitroquinoxaline-2,3,dione (DNQX; 10 μM), and bicuculine methiodide (Bicc; 20 μM). All drugs were obtained from RBI or Sigma, freshly prepared in ACSF and bath applied. Data were acquired with Labview 5.0 and a PCI-16-E1 data acquisition board (National Instrument, Austin TX.) at 10 kHz, and analyzed with Matlab.

**2.2 Stimulus Generation and Experimental Design.** To test the event finding method, we used data collected to study the effect of varying the amplitude and offset of a repeated frozen noise stimulus. The same frozen noise waveform was used for all experiments. A white noise waveform (sampling rate 10 kHz, with samples uniformly distributed on the unit interval) was generated using the Matlab function *rand* with the state of the random number generator set to zero. It was twice filtered using the Matlab routine *filter(b,a,x)*, with parameters *a* and *b*, and the white noise waveform *x*. First, we applied a low-pass filter with  $a = [1 \ -0.99]$  and  $b = 1$ . Second, we performed a 50-sample (5 ms) running average ( $a = 1$ ,  $b$  has 50 elements equal to  $1/50$ ). The first 500 samples (i.e., 50 ms) were discarded as a transient, and the resulting waveform was centered around 0, by subtracting the mean and normalized to have unit variance by dividing by the standard deviation. Depending on the cell and the quality of the seal, the waveform was multiplied by a factor representing the maximum amplitude, and an offset was added. Waveforms with fractional amplitudes from 0 to 1 were presented to the cell. The characteristics of the experimental data set are listed in Table 1. Trials were separated by at least 15 seconds of zero current to let the membrane return to its resting state. Throughout the experiment, a few hyperpolarizing pulses were injected to monitor the access resistance of the preparation. These pulses were clearly separated from other stimuli.

### 3 Analysis Procedures

---

**3.1 General Procedures.** Spike times were detected from recorded voltage traces as the time the membrane potential crossed 0 mV from below. The firing rate was the number of spikes recorded during a trial, averaged across all similar trials and normalized by the duration of the trial in seconds.

Each row of the rastergram represented a spike train from a different trial. Each spike was represented as a tick or a dot, with the spike time as the *x*-ordinate and the trial number as the *y*-ordinate. The trials were often

Table 1: Summary of the Stimulus Parameters of 11 Experiments Done on 10 Different Neurons.

Data Set	Number of Trials	Normalized Amplitudes	Maximum Amplitude	Offset	Online Gain (nA)
050501_1c	18	0:10%:100%	0.1	0.03	0.5
050501_1d (Figure 11B)	51	0:10%:100%	0.1	0.15	0.5
082003_2c (Figure 11A)	45	0:10%:100%	0.1	0.15	0.8
082503_3b	41	0:10%:100%	0.1	0.15	1.0
090203_1b	21	0:10%:100%	0.1	0.15	0.9
090303_1b (Figure 11C)	29	0:10%:100%	0.1	0.15	0.9
091803_1c	40	0:10%:100%	0.1	0.15	2.2
082301_1c	22	20%:8%:100%	0.1	0.15	0.4
082401_2b (Figure 11D)	18	20%:8%:100%	0.1	0.15	1.0
012007_2b	36	0:10%:100%	0.1	0.15	5
012607_1d	40	0:10%:100%	0.1	0.15	4

grouped together based on the stimulus amplitude or reordered based on which pattern they belonged too. This is indicated in each figure caption.

The spike time histogram is an estimate of the time-varying firing rate. It was obtained by dividing the time range of a trial into bins (typically 1 or 2 ms wide) and counting the number of spikes that fell in each bin across all trials. The bin count was normalized by the number of trials and by the bin width in seconds. The latter normalization ensured that a bin entry had the dimensions of a firing rate, Hz. The histogram was subsequently smoothed by a gaussian filter with a standard deviation equal to 1 bin size.

Events were detected using the procedure detailed in section 4. At the end of this procedure, all spikes were either assigned to an event or were classified as noise. The unitless event reliability is the fraction of trials on which a spike was observed during that event, and the event jitter (ms) is the standard deviation of the spike times belonging to the event. The event precision (1/ms) is the inverse of the event jitter. For a given amplitude, the reliability, precision, and jitter are defined as the event reliability, event precision, and event jitter averaged across all events.

We used three techniques to find an event: the Victor-Purpura distance, the fuzzy clustering method, and classification entropy and mutual information.

**3.2 Calculation of the VP Distance.** Briefly, the Victor-Purpura (VP) metric (Victor & Purpura, 1996) calculates the distance between two spike trains  $A$  and  $B$  by calculating the cost of transforming  $A$  into  $B$  (or  $B$  into  $A$ —the measure is symmetric). This distance is obtained as the minimum

cost of transformation under the following rules: adding or removing a spike from  $A$  costs +1 point, while sliding spikes forward or backward in time by an interval  $dt$  costs  $q$  times  $|dt|$ . The variable  $q$  represents the sensitivity of the metric to the timing of spikes and is expressed in units of 1/ms. For large  $q$  values, it is frequently cheaper to add and remove spikes than to move them. Hence, for large  $q$ , the metric is simply the number of spikes with different times between the two trains. For small  $q$  values, spike-moving transformations are cheap, leaving the majority of the metric's value to the difference in the number of spikes that must be added or removed to produce spike train  $B$ ; in the limit, the metric becomes the difference in the number of spikes in each spike train. For a set of  $N$  spike trains, the VP metric produces a symmetric  $N \times N$  matrix. The  $(i, j)$  entry of the matrix is the VP distance between the  $i$ th and  $j$ th spike trains.

**3.3 Fuzzy Clustering Algorithm.** Fuzzy  $c$ -means (FCM) was used to cluster trials into groups that had similar spike timings. FCM can be understood by first considering  $K$ -means clustering (also, but less commonly, referred to as  $c$ -means). In a  $K$ -means clustering, a number of clusters is chosen, and the objects to be clustered are assigned on a random basis to each of the potential clusters (Duda, Hart, & Stork, 2001). The name of the algorithm derives from the convention that the number of clusters is denoted by  $K$ , but here we denote it instead by  $N_c$  for notational consistency. The mean of each cluster is found by using these assignments. Then, using these means, objects are reassigned to each cluster based on which cluster center they are closest to. This process repeats until the cluster centers have converged onto stable values or a maximum iteration count is reached. This type of clustering minimizes the sum of the squared distances of the clustered objects from their cluster means. FCM functions in the same way, but rather than belonging to any particular cluster, each object  $i$  is assigned a set of normalized probabilities  $u_{ij}$  of belonging to cluster  $j$  (Bezdek, 1981). This is equivalent to minimizing a nonlinear objective function of the distances of the objects from the cluster centers, characterized by the "fuzzifier" parameter, which is set to 2. After the algorithm converges, each spike train is assigned to the cluster to which it is most likely to belong (maximizing the  $u_{ij}$  with respect to the cluster index  $j$ ). A more complete description is given in Fellous, Tiesinga, Thomas, and Sejnowski (2004).

We use FCM on the columns of the pair-wise distance matrix (see section 3.2) because similar trials will have a similar distance from all other trials and are thus represented by similar columns (Fellous et al., 2004). The computational effort of FCM increases with the number of vectors ( $N_{trial}$ ), as well as the dimensionality of the vectors (also  $N_{trial}$ ). Hence, we reduced the dimensionality from  $N_{trial}$  to 10 components using principal component analysis (PCA) (Jolliffe, 2002). These components accounted for at least 80% of the variance and resulted in clusterings that were similar to those obtained using all principal components.

**3.4 Calculation of Entropy and Mutual Information Between Classifications.** The outcome of the FCM procedure is that each spike train is assigned to a cluster. Formally, a set of trials has a classification  $c_i$ , where  $i$  is the trial index between 1 and  $N_{trial}$ ; and the classification  $c$  is a number between 1 and the number of clusters  $N_c$ . The class distribution  $p_j$  is the fraction of trials classified as class  $j$ ,

$$p_j = \frac{1}{N_{trial}} \sum_{i=1}^{N_{trial}} \delta(j - c_i),$$

where  $\delta$  denotes the Kronecker delta. The diversity of the classification was characterized by the entropy

$$S = - \sum_{j=1}^{N_c} p_j \log_2 p_j,$$

taking the sum over all nonzero  $p_j$  values (equivalently,  $0 \log 0$  was defined to be 0; Cover & Thomas, 1991). The entropy  $S$  was 0 (minimal diversity) when all trials were assigned to the same class (the only nonzero  $p_j$  is 1, yielding  $S = 0$  because  $\log 1 = 0$ ) and was maximal at  $S = \log_2 N_c$  when all classes had the same probability of occurring (maximal diversity). The entropy was also calculated for continuous values, such as VP distances between pairs of spike trains, in which case the same formulas were applied to a binned probability distribution. It is well known that estimation of entropy from a binned distribution is biased (Panzeri & Treves, 1995; Strong, Koberle, de Ruyter van Steveninck, & Bialek, 1998). However, because these values were used for comparative purposes, no bias correction was needed.

The mutual information was used to measure the similarity between pairs of spike train classifications. The joint distribution between two classifications  $c_i$  and  $d_j$  with  $N_c$  and  $N_d$  classes, respectively, was computed as

$$p_{ij} = \frac{1}{N_{trial}} \sum_{k=1}^{N_{trial}} \delta(c_k - i) \delta(d_k - j).$$

The mutual information was then expressed as

$$I = S_c + S_d + \sum_{i=1}^{N_c} \sum_{j=1}^{N_d} p_{ij} \log_2 p_{ij}, \quad \text{with}$$

$$S_c = - \sum_{i=1}^{N_c} p_i^c \log_2 p_i^c \quad \text{and} \quad S_d = - \sum_{j=1}^{N_d} p_j^d \log_2 p_j^d.$$

In these formulas, the class distributions for  $c$  and  $d$  have a different subscript in order to distinguish them. Hence, here,  $i$  refers to the classification rather than being a trial index. To obtain a measure between 0 and 1, we normalized the mutual information by the maximum entropy, yielding the normalized mutual information  $I_N = I/\max(S_c, S_d)$ .

## 4 Results

---

### 4.1 Finding Spike Patterns and Determining the Event Structure in Artificial Data.

*4.1.1 Overview and Goal of the Event-Finding Procedure.* In a previous study we uncovered multiple spike patterns in trials obtained in response to repeated presentation of the same stimulus (Fellous et al., 2004; Reinagel & Reid, 2002). Spike patterns were present when trials, or at least short segments thereof, could be separated into two (or more) distinct groups of spike sequences. As an example, consider the case where on some trials, the neuron spikes at 10 ms and 35 ms (relative to stimulus onset), whereas on other trials, it spikes at 15 ms and 30 ms. This group of trials would be considered as comprising two distinct patterns as long as there are no trials with spikes at 15 ms and 35 ms or 10 ms and 30 ms. Hence, spike patterns correspond to a within-trial correlation, because in the example, a spike at 10 ms implies that a spike will be found at 35 ms with a high probability. The problem of finding spike patterns is made harder by the presence of spike time jitter and trial-to-trial unreliability.

We designed a method to uncover patterns independent of the event structure and then used the patterns to construct the event structure, which can subsequently be used to validate the extracted patterns. The method itself is unsupervised, but four parameters need to be provided:

1. The temporal resolution at which two spike times were considered similar (parameter:  $q$ , section 4.1.2)
2. The number of patterns the clustering algorithm looked for (parameter  $N_c$ , which stands for the number of clusters, section 4.1.3)
3. The threshold for finding events (parameter  $t_{ISI}$ , section 4.1.4)
4. A threshold  $t_{ROC}$  that determined which events needed to be merged because they were common to multiple spike patterns (section 4.1.5)

Figure 2 provides a schematic overview of the full procedure. Readers interested in the application of the method but not in the details corresponding to each of the displayed boxes are advised to skip to Figure 12, where the procedure is illustrated using experimental data, and to return to the specific section as guided by Figure 2 as needed.

For the parameter settings used here, each cluster corresponded to a spike pattern; hence, we will use these terms interchangeably. The procedure was tested on short temporal segments with approximately two or three spikes

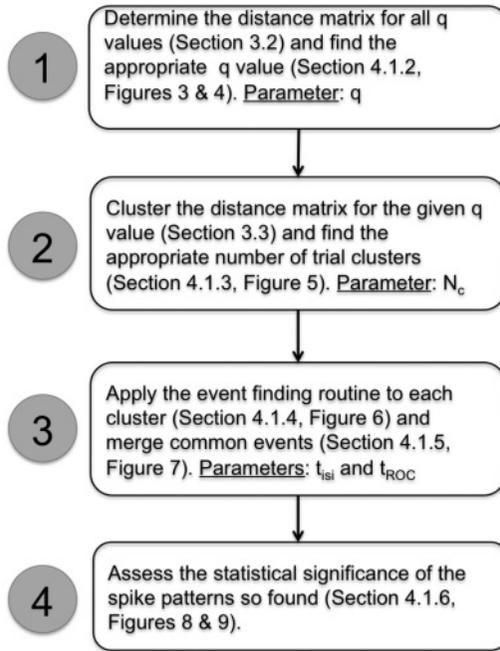


Figure 2: The different steps of the analysis procedure with the corresponding section and figure numbers.

per trial on average. These segments were found by cutting spike trains at times with a low or zero spike rate in the spike time histogram. In the following text, we restrict the discussion to a single such segment.

Heuristically, two trials on which the same pattern was produced will be more similar to each other than two trials on which different patterns were produced. We used the VP distance (Victor & Purpura, 1996; see section 3.2) to quantify the similarity. The distance between spike trains  $i$  and  $j$  is represented as a matrix  $d_{ij}$ . The distance matrix generally appeared unstructured when the trials were arranged in the order in which they were recorded (look ahead to Figure 12B for an example). The goal is to reorder the matrix such that it becomes blockdiagonal (see Figure 12F). The blocks then correspond to trials that have a small distance among themselves and are dissimilar to trials outside the block. That is, blocks on the diagonal correspond to spike patterns.

This goal was achieved using the FCM method (Bezdek, 1981) applied to the columns of the distance matrix (see section 3.3). FCM found  $N_c$  clusters and assigned to each trial a probability of belonging to a cluster. If each trial belonged to only one cluster with a high probability, it was considered a good trial, but if a trial had similar probabilities of belonging to

two or more different clusters, it was considered a bad trial. Once patterns had been uncovered a preliminary event structure was determined using the method outlined in section 4.1.4 on each pattern (Tiesinga et al., 2002). The common events that occur in multiple spike patterns were found and merged using a receiver operating characteristic (ROC) analysis (Green & Swets, 1966) as outlined in section 4.1.5. After this analysis, a cluster-assisted event structure was available. For the purpose of this article, the spike pattern detection and event assignments for both simulation and experimental data were manually checked and corrected using an in-house interactive software tool. We found that the results obtained using the unsupervised method on artificial data matched those found manually in most of the cases where there was a clear event structure. For the 474 segments of experimental data from 10 cells analyzed in this fashion (see section 4.2 for examples), we found that 10% did not have a clear event structure and 45% had some recognizable events together with parts that did not appear to consist of events. For the remaining segments, the semiautomatic analysis differed by two or fewer mergers of events compared to the manual analysis, which was considered adequate agreement. The event structure can be used to determine the significance of spike patterns, according to the method presented in section 4.1.6.

*4.1.2 Selecting an Appropriate Temporal Resolution Parameter  $q$ .* The first step in uncovering spike patterns was to select an appropriate value for the temporal-resolution-parameter  $q$ . The defining characteristic of spike patterns was a significant difference in event timing. Furthermore, spikes in spike patterns could be unreliable, which means there could be a difference in spike count between trials expressing the same spike pattern. Hence, low  $q$  values were not appropriate because in this case, the distance would be dominated by the difference in spike count, which would split up patterns with unreliable events. Note also that for low  $q$  values, the elements  $d_{ij}$  of the distance matrix were approximately integer. High  $q$  values were also not appropriate because the distance would be dominated by small differences in spike timing, which would split trials expressing the same pattern when events were imprecise. For large  $q$  values, the distance approached the difference in spike count plus twice the number of spike pairs that were different by more than  $2/q$ ; therefore, the distance again took values close to integers. (Note that the distance was not exactly an integer because there were still some spike pairs separated by less than  $2/q$ , although their number indeed went to 0 in the limit where  $q$  goes to infinity.) The optimal  $q$  value was between these two extremes. We developed a procedure to find this value using only information on how the ensemble of distances  $d_{ij}$  behaved as a function of  $q$ .

The procedure was tested using artificially patterned spike trains mimicking experimental data (see Figure 3A). The first step was to combine all the matrix elements (see Figure 3B) into one set and calculate a histogram

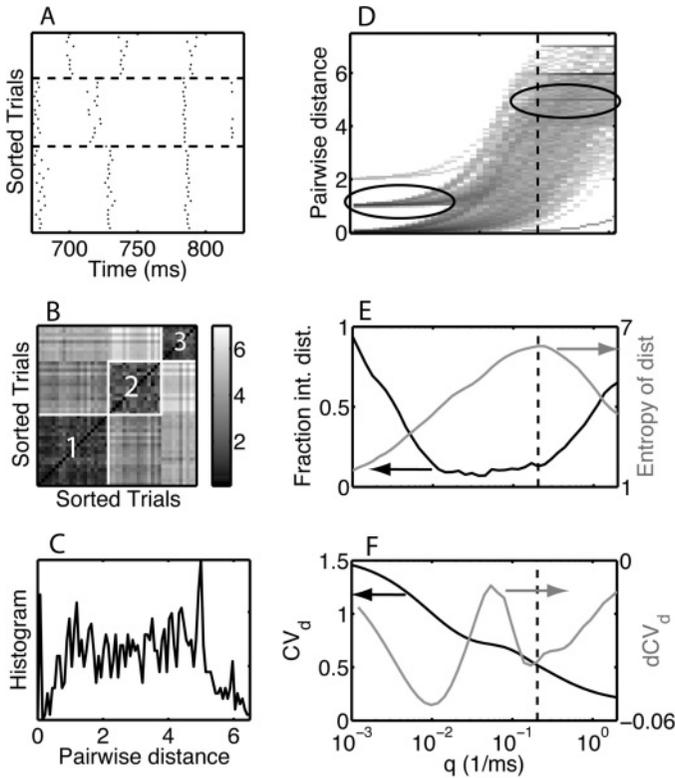


Figure 3: Heuristic for selecting the temporal resolution parameter  $q$ . (A) Rastergram of 45 trials containing three patterns (separated by horizontal dashed lines). (B) The distance matrix for the below-determined  $q$  value. The matrix has a diagonal structure with blocks of small distance values corresponding to the patterns (as labeled by numbers inside the blocks). (C) The histogram of the matrix elements of the distance matrix shown in panel B. (D) Density plot of the distance distribution as a function of  $q$ . Each column represents a histogram just like the one shown in panel C, the location of which is indicated by a vertical dashed line. The ellipses highlight examples of enhanced density for integer values for the distance for small and large values of  $q$ . Panels D–F have a common  $x$ -axis scale, which is shown in panel F. (E) We show (black curve, left-hand-side scale) the fraction of distances within 0.05 from an integer and (gray curve, right-hand-side scale) the entropy of the distance histogram as a function of  $q$ . (F) We show (black curve, left-hand-side scale) the coefficient of variation of the distances ( $CV_d$ ) and (gray curve, right-hand-side scale) the differenced  $CV_d$  ( $dCV_d$ ). The value of  $q$  is selected based on the peak of the entropy (panel E) and the first minimum following the peak of the  $dCV_d$  (panel F). See the text for a detailed description of the heuristic procedure.

(see Figure 3C). A new matrix was obtained by stacking the histograms for each  $q$  value as columns (see Figure 3D). The image of this matrix showed a transition from integer-dominated distances for low  $q$  values (see Figure 3D, left ellipse) to integer-dominated distances for high  $q$  values (see Figure 3D, right ellipse) via a broader and a less peaked transition state. We found that the best results were obtained by using  $\log(q)$  values uniformly distributed between  $10^{-4}$  and 2 (1/ms). The transition could be visualized by determining the fraction of distances within 0.05 of an integer as a function of  $q$  (see Figure 3E, black line, left scale). As expected, a trough emerged at intermediate  $q$  values. Because the entropy of a distribution with a sharp peak was lower than that of a broader distribution, the transition was signaled by a peak in the entropy of the distance distribution as a function of  $q$  (see Figure 3E, gray line, right scale). The entropy was calculated using the formula

$$S_d = - \sum_i h_i \log_2 h_i,$$

where subscript  $d$  means distance and  $h_i$  is the fraction of distance values falling in the  $i$ th bin of the distance histogram (see section 3.4).

We achieved good results using 200 bins to cover distance values obtained across the entire range of  $q$  values studied. A more sensitive measure for detecting changes in the distinguishability of patterns was based on the coefficient of variation of the distances ( $CV_d$ ), which, unlike the entropy measure, did not require binning of the data. The  $CV_d$  generally decreased with  $q$  (see Figure 3F, black line, left scale), but there were modulations in the rate of change, which reflected the structure in the data that we sought to uncover. These modulations were clearest in the differenced  $CV_d$  ( $dCV_d$ ; see Figure 3F, gray line, right scale). This measure corresponded to the derivative of  $CV_d$  with respect to  $\log q$ , because the  $\log q$  values were uniformly distributed. We found that the best clusterings were obtained at the  $q$  value corresponding to the deepest trough after the highest peak. However, this location varied across independent stochastic realizations of spike trains generated using the same spike pattern parameters (see sections 4.1.6 and 4.1.7). Hence, to make the procedure more robust, we choose the  $q$  value to be the mean of the position of the peak in the entropy  $S_d$  (less sensitive but robust) and the position of the deepest trough after the highest peak in the  $dCV_d$  (sensitive but less robust). The resulting  $q$  value is denoted by  $q_{\max}$ . Note that the mean of the peak and trough was not exactly in the middle when the data are plotted on a log scale for  $q$  as in Figures 3E and F.

To gain more insight into the sensitivity of the  $dCV_d$  modulation to features of the spike patterns, we analyzed the clusterings obtained for different  $q$  values. For this analysis, we used the best clustering (optimal  $N_c$  value) determined by the differenced gap statistic defined in section

4.1.3. A clustering assigns to each trial  $i$  a class  $c_i$ , which is also referred to as a classification. Two clusterings were compared by computing the normalized mutual information  $I_N$  between two classifications (see section 3.4). When the two clusterings were identical,  $I_N = 1$ . This is true even when the class labels were different (e.g., when class 1 in classification  $c$  corresponds to class 2 in classification  $d$ ), because the mutual information was insensitive to a permutation of class labels.

We constructed a similarity matrix  $SM_C$  using the normalized mutual information  $I_N$  between clusterings obtained for different  $q$  values. Specifically, the element in the  $i$ th row and  $j$ th column is the  $I_N$  between the clustering for the  $i$ th  $q$  value and the clustering for the  $j$ th  $q$  value (both with the correct number of clusters  $N_c$ ). The resulting matrix (see Figure 4A) consisted of diagonal blocks, indicating that the same clustering was obtained over a range of  $q$  values: the larger the block, the more robust the clustering. Because the data were generated from a known parameter set, the correct class assignments were known. Hence, the mutual information between the best clustering for a specific  $q$  value and the correct classification were determined. The resulting graph of  $I_N$  versus  $q$  (see Figure 4B) consisted of a sequence of steps, each corresponding to a distinct clustering that was obtained across a range of  $q$  values. Transitions between clusterings were related to modulations in the  $dCV_d$  (see Figure 4C). The correct clustering was obtained just after the highest peak in  $dCV_d$  (see Figure 4C; white arrow in panel A) but only for a small range of  $q$  values. The next best clustering was obtained in a broad range of  $q$  values around the deepest trough following the highest peak in the  $dCV_d$  (see Figure 4C, asterisk in panel A). This clustering was selected by the heuristic. Although for experimental data, the correct clustering was not available, the similarity matrix between clusterings for different  $q$  values could still be determined in order to select a small number of distinct and robust clusterings for a more detailed evaluation.

4.1.3 *Selecting the Number of Patterns/Clusters  $N_c$ .* Once an appropriate  $q$  value was selected, the optimal number of clusters  $N_c$  needs to be determined. This is a difficult problem that has attracted significant attention (Bouguessa, Wang, & Sun, 2006; Pal & Bezdek, 1995; Rezaee, Lelieveldt, & Reiber, 1998; Zahid, Limouri, & Essaid, 1999). Intuitively, the best clustering minimized the distance between trials within a cluster while maximizing the distance between trials in different clusters. The output of the FCM algorithm was the matrix  $u_{ij}$ , which was the probability of trial  $i$  belonging to cluster  $j$  and the cluster centers, which were obtained by averaging across all vectors corresponding to trials in the cluster. A large number of cluster-validation measures have been proposed based on  $u_{ij}$  and the cluster centers that quantify these intuitive expectations (Bouguessa et al., 2006). An estimate for the number of clusters was obtained from the number that minimized (or maximized, as appropriate) these validation measures. In

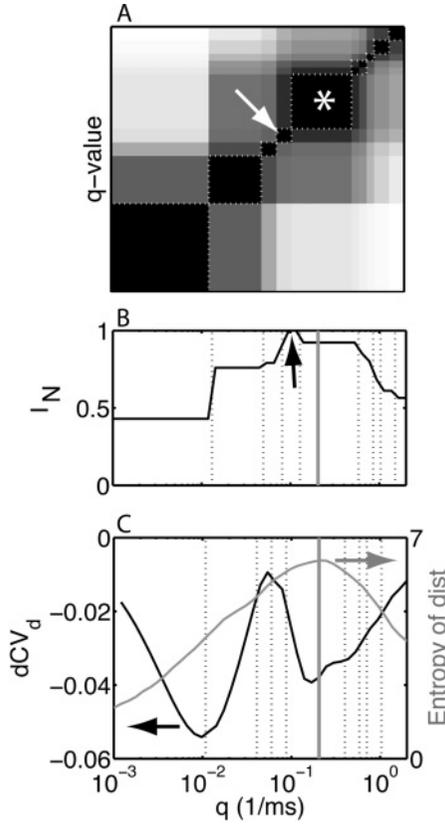


Figure 4: The  $q$  value chosen by the heuristic corresponds to a robust close-to-optimal clustering. We analyzed an artificial data set with three patterns and a known classification. The FCM algorithm was applied to distance matrices for different  $q$  values, with the number of clusters  $N_c$  set to three. (A) The pairwise similarity between clusterings (classifications) for different  $q$  values was quantified using the normalized mutual information ( $I_N$ ; see section 3.4) and is shown as an image. The range of  $I_N = 0$  to  $I_N = 1$  is represented on a gray scale from white to black. The clustering indicated by the arrow was optimal; the clustering picked by the  $q$ -value heuristic is part of the block labeled by an asterisk. The blocks of similar clustering are delimited by dashed boxes, which were obtained by clustering the classification similarity matrix. (B) The similarity between the known classification and those obtained by the FCM algorithm as a function of  $q$ . The black arrow indicates the correct clustering. (C) We show (black curve, left-hand-side scale) the  $dCV_d$  and (gray curve, right-hand-side scale) the entropy of the distances as a function of  $q$ . The dashed vertical lines in (B & C) correspond to the dashed boxes in panel A. The vertical gray line is the  $q$  value chosen by the heuristic. The  $x$ -axis scale is the same for panels A to C and is shown in panel C.

previous work, these measures were successfully tested on data generated from gaussian mixture models (Bouguessa et al., 2006). These measures did not generalize well on our experimental data, which did not satisfy the assumptions for a gaussian mixture model (see below).

Our criterion for picking the number of clusters was instead based on the gap statistic (Tibshirani, Walther, & Hastie, 2001; Yan & Ye, 2007), which was calculated as follows. The data were divided into  $N_c$  clusters with cluster assignments  $c_k$ . The within-cluster distance  $D_r$  for the  $r$ th cluster was the Euclidean distance  $d_{kk'}$  summed across pairs of trials in the same cluster,

$$D_r = \sum_{k,k'} \delta(r - c_k) \delta(r - c_{k'}) d_{kk'}.$$

Note that the Euclidean distance was based on the first 10 principal component coordinates of columns in the distance matrix (see section 3.3) because the spike trains themselves were not vectors in a Euclidean space (Victor & Purpura, 1996). The weighted distance for  $N_c$  clusters was

$$W_{N_c} = \sum_{r=1}^{N_c} \frac{D_r}{2n_r},$$

where  $n_r$  is the number of elements in cluster  $r$ , that is,

$$n_r = \sum_{k=1}^{N_{\text{trial}}} \delta(r - c_k).$$

The quantity  $W$  was compared to values of clusters obtained from clustering surrogate data that were uniformly distributed in the range spanned by the original data. Denoting the weighted distance for  $N_c$  clusters obtained for the  $i$ th surrogate data set by  $\tilde{W}_{N_c}^i$ , the gap statistic was

$$G_{N_c} = \frac{1}{B} \left( \sum_{i=1}^B \log(\tilde{W}_{N_c}^i) \right) - \log(W_{N_c}),$$

where  $B$  is the number of surrogates used.

When data with  $C$  clusters were generated using a gaussian mixture model, this measure correctly peaked at  $N_c = C$ . For the artificial data generated based on our experimental recordings, this was generally not the case, because a cluster with only one pattern but with an unreliable event could always be broken up into two clusters, yielding smaller within-cluster distances. For example, consider the case for which event 2 is unreliable.

The cluster containing the pattern can be split into a cluster without a spike during event 2 and a second cluster with a spike during event 2. We found that for an appropriate  $q$  value, the largest change in the gap statistic occurred when a cluster containing two distinct patterns was split in two (see Figure 5A). We therefore used the following heuristic. We pick the number of clusters obtained after the largest increase in the gap; that is the one yielding the largest value of the gap difference,  $dG_{N_c} = G_{N_c} - G_{N_c-1}$  (asterisk in Figure 5A). For our data set, we added two more rules to the heuristic. We found them to be appropriate for our data, but we did not evaluate their performance on a broader set of data. Hence, for other data sets, the percentages quoted in the following should be adjusted as needed. First, when there were two or more nearly equal consecutive increases (here, to within 20%), we pick the cluster number obtained after the latest increase, even if it was the smaller increase. Second, if the first  $dG$  value was the highest but there was another peak a certain fraction (here, 80%) of the highest  $dG$  value, this peak was chosen instead.

The heuristic for choosing  $q$  and  $N_c$  was studied using surrogate data sets with varying numbers of spike patterns and trials. For Figure 5, we generated 10 independent sets of 45 trials containing the same three spike patterns. For each  $q$  value, the distance matrix was first calculated, then clustered 10 times for  $N_c$  values between 2 and 10. For each  $N_c$ , the clustering with the lowest  $W_{N_c}$  value was chosen. Subsequently, the gap statistic was determined by clustering  $B=10$  surrogate data sets. The number of clusters that maximized  $dG$  was picked (for these data, there was only one peak). The clusterings so chosen were compared to the correct clustering in terms of the normalized mutual information  $I_N$ . In Figure 5B, the number of clusters chosen is shown as a function of  $q$ . In the same graph, the heuristic choice of  $q$  for each set of trials is indicated by a tick, and the mean  $q$  across 10 sets is indicated by the dashed vertical line. This mean  $q$  value is in the middle of the range of  $q$  values for which the  $dG$  statistic picks the correct number of three clusters or patterns with 100% accuracy. For other  $q$  values, there was a high variability in the number of clusters selected, and the mean was different from the correct value. In Figure 5C,  $I_N$  is plotted as a function of  $q$ . It has a peak value near 1, whose location is correctly predicted by the heuristically chosen value of  $q$ .

To evaluate how the performance of this procedure generalized to other patterns, we determined three statistics: (1) how often the correct number of clusters was picked, (2) the mean value of the normalized mutual information  $I_N$  at the selected  $q$  and  $N_c$  value, and (3) the relative height of the peak in the difference gap statistic,  $dG$ . The relative height was obtained by comparing the peak  $dG$  to the mean and standard deviation of the other  $dG$  values with the peak value excluded. It was numerically equal to the peak in  $dG$  minus the mean, after which the result was divided by the standard deviation. Values exceeding 2 were considered satisfactory, but we did not evaluate the statistical significance of this threshold. We used two data sets

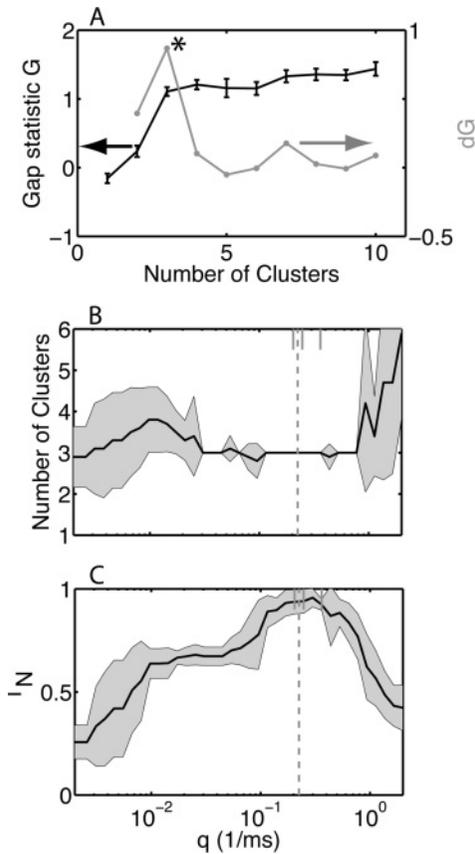


Figure 5: The heuristics for finding the  $q$  value and the number of clusters perform well for artificial data sets. (A) We show (black curve, left-hand-side scale) the gap statistic  $G$  and (gray curve, right-hand-side scale) its difference  $dG$  as a function of the number of clusters. The number of clusters  $N_c$  reached after the highest increase in the gap statistic (at the peak of  $dG$  indicated by the asterisk) is selected as a heuristic for the number of clusters in the data. (B) We generated 10 independent artificial data sets of 45 trials with three patterns, all of which had the same statistics as those shown in Figure 3. The average  $N_c$  value picked by the heuristic is plotted as a function of  $q$ ; the shaded area represents the standard deviation. (C) The similarity  $I_N$  between the clustering obtained with the selected  $N_c$  value and the known classification as a function of  $q$ . The black line is the mean, and the shaded area represents the standard deviation. The dashed line in panels B and C is the mean value of the  $q$  value picked by the heuristic; the choice for each of the 10 data sets is indicated by the ticks at the top of the graph (these ticks are restricted to be on the log-uniform distributed  $q$  values). The heuristically picked  $q$  value leads to an optimal similarity and an  $N_c$  that is equal to the actual number of clusters present in the data.

that were based on analyzed experimental data (the first one, E1, had been used for the data in Figures 3, 4, and 5) and an artificial set constructed to have four patterns. The procedure performed well on all these sets as measured by the three statistics. Specifically, the correct number of clusters was picked between 90% and 100% of the time;  $I_N$  exceeded 0.90, and the relative peak in the gap was higher than 3. The effect of the number of available trials on the performance of the entire procedure is investigated in section 4.1.7.

*4.1.4 The Interval Method for Identifying Events.* Our event-finding procedure was based on the interspike intervals (ISI) of an aggregate spike train, which was obtained by combining spikes across all trials into one set and sorting them, with the earliest spikes first (Tiesinga et al., 2002). Intuitively, consecutive spikes in the same event were close together and had a small ISI, whereas those in different events were farther apart (see Figure 6A). We could thus separate spikes in different events using a threshold  $t_{ISI}$ : only spikes separated by an interval less than  $t_{ISI}$  were considered to be in the same event. In addition, we required that each event contains more than  $m_s$  spikes. Taken together, this led to the following algorithm:

1. Take all spikes from all trials and place them in an aggregate, time ordered list.
2. Beginning with the first spike, collect spikes (and remove them from the list) until the difference between the last collected spike and the next spike is greater than  $t_{ISI}$ .
3. If the collected spikes exceed the minimum number of spikes required for an event ( $m_s$ ) assign them to a new event. Otherwise, categorize the spikes as “noise.”
4. If any spikes remain in the time-ordered list, return to step 2.

We set a minimum spike number for two reasons. First, we wanted to avoid calculating the precision based on too few spikes. Second, we sought to avoid interpreting a near coincidence of two or more noise spikes as an event. To see how a near coincidence could happen, consider the following hypothetical situation where the data set is generated by a Poisson process with a fixed rate of  $R_{noise}$ . The aggregate spike train is then a Poisson process with a rate  $R = N_{trial} R_{noise}$ . The probability  $p$  of obtaining two spikes with a distance of less than  $t$  is  $p = R \int_0^t \exp(-Rs) ds = (1 - \exp(-Rt))$  (Rieke, Warland, de Ruyter van Steveninck, & Bialek, 1997). Because  $R$  increases with the number of trials, for a large enough number of trials, there likely will be near-coincident noise spikes. Specifically, for  $R_{noise} = 1$  Hz and  $N_{trial} = 100$ , the probability of a within 3 ms coincidence is 26%. In the analysis performed for this article, between 10 and 150 trials were used and the  $m_s$  was set to 2.

The correct threshold for interspike intervals was determined by three distinct timescales in the data, which are shown in Figure 6. An

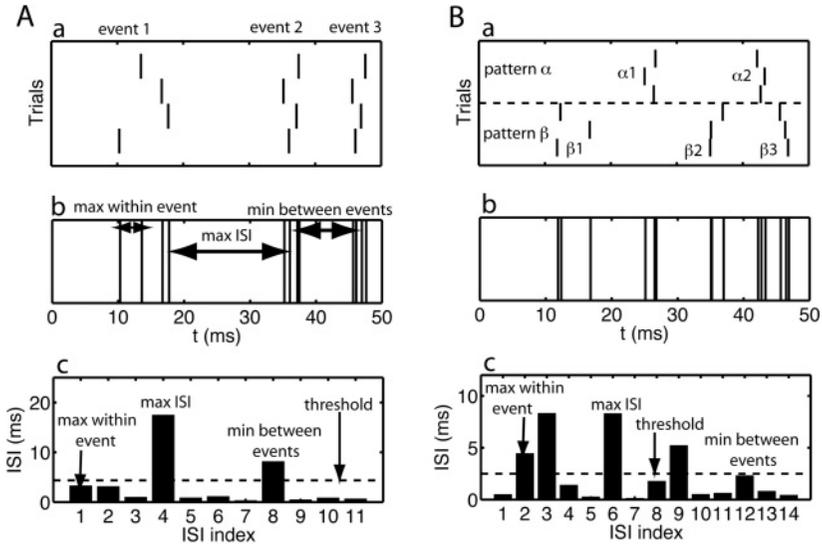


Figure 6: It is not always possible to choose an appropriate threshold for the interval method. (A–B) In each panel, we show, from top to bottom, (a) the rastergram, (b) the aggregate spike train, and (c) the ISI time series. The relevant timescales are indicated in the graph: the maximum ISI between two spikes in the same event, the minimum ISI, and the maximum ISI between two spikes in different events. An estimate for the threshold, 0.1 times the maximum ISI, is shown as a dashed line in panel c. (A) An example with one spike pattern where an appropriate threshold can be chosen that is higher than the maximum ISI within an event but less than the minimum ISI between events. (B) An example with two spike patterns (labeled by  $\alpha$  and  $\beta$ ) where there is no such threshold that separates within-event intervals from those between events.

appropriate threshold was greater than all ISIs within an event but less than all ISIs between spikes in different events (see Figure 6A). Because two consecutive spikes in the aggregate spike train came from different trials, the interevent interval can be smaller than the refractory period, which means that in some cases, no value for the threshold satisfied the constraint (see Figure 6B). This problem was addressed by grouping trials according to the spike patterns they expressed and analyzing each group separately with the interval method. We found that a threshold value of 10% of the maximum ISI often provided results similar to events detected by hand. As a rule of thumb, the appropriate value for the threshold decreased as the number of available trials increased. For the data sets in this article, we used a  $t_{ISI}$  between 1 and 3 ms.

*4.1.5 Merging Events Common Across Multiple Patterns.* The interval method for event detection may lead to separate identification of events for individual spike patterns even when those events were common to multiple patterns. (See Figure 12E for an example from experimental data.) These common events had to be merged prior to assessing the significance of the patterns. The  $t$ -test is a statistical procedure for testing whether the means of two groups of data are equal under the assumption that their standard deviations are the same (Larsen & Marx, 1986; Rohatgi, 2003). The means (events) were considered different when the probability of obtaining a  $t$ -value higher than the measured value was less than the significance level.

Within the context of the experimental data, it was not possible to use this statistic to design an automatic method for merging events. First, the  $t$ -statistic tests whether the samples are drawn independently from one normal distribution. For the experimental data, events were often split across patterns in a nonsymmetric way. For instance, for spike times in a common event, it was often the case that the spikes belonging to one pattern always appeared before those of the other pattern. This could be modeled by drawing spikes from a normal distribution, sorting them, and assigning the lowest half to one group and the remainder to the second group. Because the spikes in these groups were not independently drawn, the difference in means between the two groups would be highly significant according to the  $t$ -statistic even though all spike times came from the same distribution. Second, even if there was a well-defined criterion for pair-wise merging of clusters, we found that a problem arose. Consider three groups of spikes. As groups 1 and 2 were merged, the standard deviation of the aggregate group would likely increase. This increased the likelihood of the combined group's merging with the third group, even though by itself, group 1 would not merge with group 3. Hence, the result depended on the order in which the pair comparisons were made. Our goal was to design a procedure that was characterized by only a few parameters that would yield reproducible results while addressing these preceding two issues.

The  $t$ -statistic measures the significance of the difference in means over the standard deviation ( $\Delta m/std$ ). We used a ROC analysis for the same purpose (Green & Swets, 1966), because it had the advantage that it did not assume an underlying gaussian distribution. Assume that there are two sets of spike times described by probability distributions  $p_1(x)$  and  $p_2(x)$ , respectively, with the mean of the first below the mean of the second (see Figure 7A). A possible decision rule is to choose a threshold parameter  $\mu$  and assign an observation  $x$  to  $p_1$  if  $x < \mu$  and to  $p_2$  otherwise. The true positive rate of this rule is  $1 - \varepsilon_1 = \int_{-\infty}^{\mu} p_1(x) dx$ , and the false positive rate is  $\varepsilon_2 = \int_{-\infty}^{\mu} p_2(x) dx$ . The ROC curve is  $1 - \varepsilon_1$  plotted versus  $\varepsilon_2$  for all possible choices of the threshold  $\mu$ . Because the theoretical distribution was not available, we used the empirical distribution with a delta function at

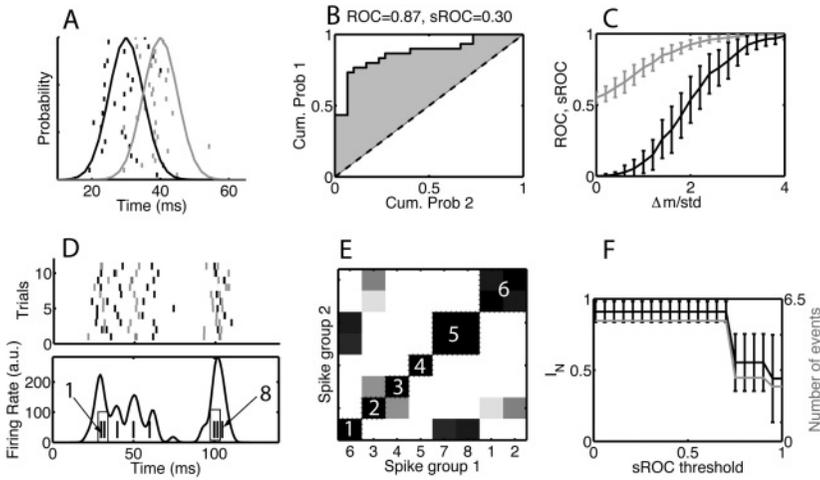


Figure 7: The ROC-based criterion for merging events common across multiple patterns. (A) Thirty trials of (black ticks) spikes drawn from a (black curve) gaussian with a mean of 30 ms and 30 trials of (gray ticks) spikes drawn from a (gray curve) gaussian with mean 40 ms. The standard deviation of the gaussian was 5 ms for both cases. (B) The ROC curve was estimated as the cumulative distribution of the first group of spikes plotted versus that of the second group of spikes (see section 4.1.5). The ROC area was 0.87. The scaled ROC (sROC) was 0.30, which was obtained by normalizing the ROC area above the diagonal by 0.5 and taking the fourth power. (C) The same procedure was repeated for different values of the difference in means between the two distributions. We show the (gray curve) ROC and (black curve) sROC area versus the difference of theoretical means over the standard deviation ( $\Delta m/\sigma$ ). We use the sROC because it is more linear near  $\Delta m/\sigma = 2$ . (D) Eleven trials of an example data set consisting of eight events, of which two sets should be merged based on the criterion  $\Delta m/\sigma < 1$ . In panel D, (top, rastergram) the events are sorted by their theoretical means, with the corresponding spikes rendered alternately in gray and black. Bottom: The histogram obtained from one realization of the data set. The ticks at the bottom represent the theoretical means of the events and are labeled 1 to 8, with the events to be merged shown in a box. (E) Matrix of sROC distances between the events. The events are ordered (the numbers on the  $x$ -axis correspond to those in panel D) according to a hierarchical clustering with an sROC threshold of  $t_{ROC} = 0.5$ . (F) Black curve: The normalized mutual information between the merged events obtained using the sROC criterion and the desired merging. Gray curve: The number of events remaining after merging. The error bars on the mutual information represent the standard deviation across 100 independent realizations of the data set. For the event distribution shown here, the desired merges are not always achieved, because sometimes events 6 and 7 are merged and sometimes events 7 and 8 are merged.

the location of each observation:  $p_1(x) = \frac{1}{N} \sum_{i=1}^N \delta(x - x_i)$ . Here,  $N$  is the number of observations  $x_i$  drawn from the first distribution. An analogous definition holds for  $p_2(x)$ . The area under the ROC curve ranged from 0.5 for distributions that cannot be distinguished to 1 for distributions that could be perfectly distinguished. We show two distributions in Figure 7A, together with 30 samples drawn from them. The resulting ROC curve is in panel B, with the area above the diagonal shown in gray. We used a scaled ROC value defined as

$$\text{sROC} = \left( \frac{\text{ROC area} - 0.5}{0.5} \right)^4$$

because we found it to have a larger dynamical range (between 0 and 1) and to be more linear around 0.5. The ROC and sROC are compared in Figure 7C as a function of  $\Delta m/\text{std}$ . The data set shown in Figure 7D contained more than two events. For each pair of events, the sROC distance was determined and placed in a distance matrix (see Figure 7E). We clustered this matrix using a hierarchical clustering method (using the Matlab routines *linkage* followed by *cluster*) to merge all groups of events with a pair-wise sROC distance less than the threshold  $t_{\text{ROC}}$ . Note that events were not merged pair-wise; all events were merged at once, after which their statistics (standard deviation, reliability) were recalculated. We analyzed the performance of this procedure for multiple independent realizations of the data sets. The procedure performed well even for sets with as few as 11 samples per event, and it was relatively insensitive to the value of the sROC threshold (see Figure 7F). Typically we used sROC thresholds of about 0.5.

**4.1.6 Binary Word Representation of Spike Patterns.** Once the event structure has been determined, the data were represented as binary words for the purpose of validating the presence of spike patterns. The spike time jitter was ignored for this analysis. Each trial was a binary word with a length (the number of bits) equal to the number of events that were detected. Each bit was 1 when there was a spike during the event and 0 otherwise. The entire experiment was reduced to a matrix  $b_{ij}$ , with  $i$  being the trial index (from 1 to  $N_{\text{trial}}$ ) and  $j$  being the event index (from 1 to  $N_E$ ). The null hypothesis was that each event is independently occupied with a probability  $p_j$ , the reliability of the event. The event reliability was estimated as the fraction  $r_j$  of trials on which there was a spike during the event, specifically

$$r_j = \frac{1}{N_{\text{trial}}} \sum_{i=1}^{N_{\text{trial}}} b_{ij} \quad \text{and} \quad p_j = \langle r_j \rangle,$$

where the average is taken across experiments with the same number of trials. The probability for obtaining a word

$$w = [b_1 \cdots b_{N_E}] = \sum_{j=1}^{N_E} b_j 2^{j-1}$$

under the null hypothesis is a product of Bernoulli probabilities,

$$P_{null}(w) = \prod_{j=1}^{N_E} p_j^{b_j} (1-p_j)^{(1-b_j)}.$$

When there are two (or more) spike patterns, the experiment was described as a mixture model of two (or more) word distributions. For simplicity, we describe the procedure assuming that there were only two patterns: A and B. This yielded  $P_{patt}(w) = c_A P_A(w) + c_B P_B(w)$ , with  $c_A$  being the fraction of trials with pattern A, and  $c_B = 1 - c_A$ ,

$$P_A(w) = \prod_{j=1}^{N_E} p_{A,j}^{b_j} (1-p_{A,j})^{(1-b_j)} \quad \text{and} \quad P_B(w) = \prod_{j=1}^{N_E} p_{B,j}^{b_j} (1-p_{B,j})^{(1-b_j)}.$$

In this expression,  $p_{A,j}$  and  $p_{B,j}$  are the reliabilities for event  $j$  in pattern A and B, respectively.

Certain events in pattern A were never occupied because they belonged to pattern B. For a word containing such an event,  $P_A(w) = 0$ . In order to estimate the parameters for this pattern distribution, each trial needed to be assigned to a pattern (e.g., using the clustering procedure described in the preceding text). The reliabilities  $p_{A,j}$  were then estimated as

$$r_{A,j} = \frac{1}{N_A} \sum_{i \in A} b_{ij},$$

where the average was across all trials  $i$  on which pattern A was obtained. The pattern probability  $\hat{c}_A$  was estimated as the fraction of trials on which pattern A was obtained. Similar expressions hold for  $r_{B,j}$  and  $\hat{c}_B$ .

The pattern distribution was not necessary in order to accept or reject the null hypothesis, but we used it to determine how many trials were required to reliably detect a given pattern using estimated reliabilities. Each experiment  $b_{ij}$  corresponded to  $N_{trial}$  words  $w_j$ , which were represented by the word distribution  $n_w = \sum_{i=1}^{N_{trial}} \delta(w - w_i)$ , where, as before,  $\delta$  denotes the Kronecker delta. The significance of the pattern is determined in terms

of the empirical  $\chi^2$  computed using the estimated reliabilities  $r_j$ :

$$\chi_{emp,null}^2 = \sum_w \frac{(n_w - N_{trial} P_{null}(w | \{r_j\}))^2}{N_{trial} P_{null}(w | \{r_j\})}.$$

When the probability to obtain this  $\chi^2$  value was less than the significance level, the null hypothesis was rejected, which means there was a significant spike pattern.

We used a distribution with parameter values  $p_A = [0.4 \ 0 \ 0.8 \ 0]$ ,  $c_A = 0.5$ ,  $p_B = [0 \ 0.6 \ 0.6 \ 0.8]$ , and  $c_B = 0.5$ . The corresponding parameter value for the null hypothesis distribution was  $p = [0.2 \ 0.3 \ 0.7 \ 0.4]$ . The rastergram and binary word representation for an example experiment with pattern  $A$  and  $B$  each expressed on 10 trials is shown in Figures 8A and 8B. The corresponding theoretical reliability model is shown in Figure 8C. In Figure 8D, the empirical word distribution (black vertical lines) is compared to the theoretical pattern distribution  $P_{patt}$  (gray vertical bars), whereas in Figure 8F, it is compared to the theoretical null hypothesis distribution  $P_{null}$  (gray vertical bars). The key observation is that the empirical distribution was different from both theoretical distributions. As the number of trials increased, the  $\chi^2$  difference of the pattern spike trains with the theoretical pattern distribution was expected to decrease, whereas the difference with the null hypothesis distribution should increase.

We generated 1000 word distributions  $n_w$  each from the pattern distribution and from the null hypothesis distribution and calculated the  $\chi^2$  between the respective word distributions and the exact null hypothesis distribution. For 20 trials (see Figure 9A), the two distributions of  $\chi^2$  values obtained across 1000 realizations of the empirical word distribution were overlapping, whereas for 100 trials (see Figure 9B), they were clearly separated. The ability to distinguish these two  $\chi^2$  values based on a single experiment was quantified using an ROC analysis based on the difference between the two distributions. The area under the ROC curve increased from 0.5 (not distinguishable) as the number of trials increased. For more than 50 trials, the empirical word distribution could be perfectly distinguished from the null hypothesis distribution (area under ROC curve close to 1), but the distinguishability was better than chance as long as there were more than 10 trials.

*4.1.7 The Effect of the Number of Trials on the Heuristics for  $q$  and  $N_c$  and the Quality of the Event Assignment Procedure.* We investigated how the quality of the data analysis procedure depended on the number of trials available for analysis. An artificial data set with 4 patterns and 11 distinct events was generated, 3 patterns had 3 events each and 1 pattern had only 2 events. Each pattern had 1 event that was unreliable ( $p = 0.5$ ), whereas the rest of the events was reliable with  $p = 0.9$ . All events had the same precision,

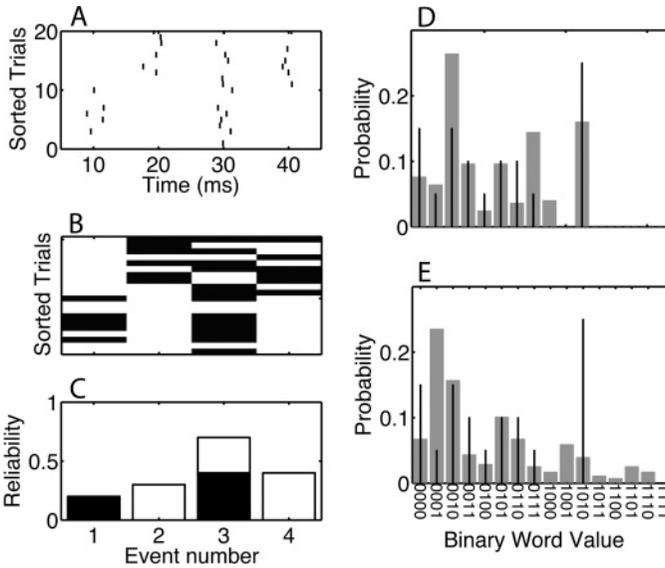


Figure 8: Binary word representation for a data set with patterns. (A) An example rastergram with 20 trials and two patterns (pattern 1 for the bottom 10 trials and pattern 2 for the top 10 trials). (B) In the corresponding binary representation, the presence of a spike during an event is indicated by a 1 (dark rectangle) and the absence by a 0. (C) The event model from which the data in panel A were obtained. The event occupation is a mixture. The event reliability for pattern 1 is indicated by the dark bars, whereas the event reliability of pattern 2 is indicated by the white-filled bars. The event model corresponding to the null hypothesis is the sum of both bars. Each binary word can be represented as a number; for four events, this number is between 1 and 16 (the binary value plus 1). Each binary word occurs with a certain probability,  $n_w$ , which is different for (D) the pattern model compared to (E) the null hypothesis model. (D, E) The theoretical probabilities are represented by the gray bars, and the empirical distribution corresponding to the data in panel B is indicated by the stem graphs. For example, the most frequent empirically occurring word is 1010, indicated by the tallest stem at the eleventh word.

which was characterized by the standard deviation  $\sigma$  (jitter), which took values between 1 ms and 5 ms. For the former precision, none of the events overlapped, whereas for the latter they did, thereby challenging the event finding procedure. We varied the number of trials  $N_{trial}$  between 8 and 256. For each parameter set  $(N_{trial}, \sigma)$ , the analysis procedure was repeated across 100 to 500 independent realizations of  $N_{trial}$  spike trains to determine how good the obtained values of (1)  $q_{max}$  and (2)  $N_c$  were and how good the (3) event assignment procedure was.

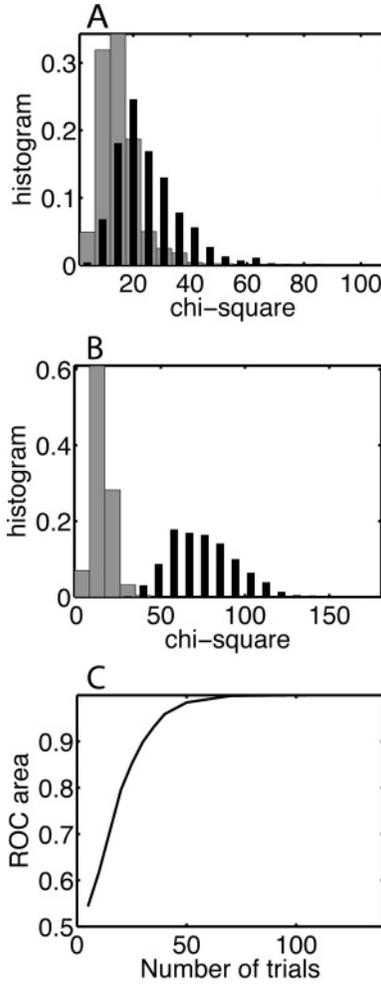


Figure 9: The difference between the patterned spike trains and the null hypothesis model can be determined perfectly when more than 50 trials are available. The chi-square statistic is calculated for the difference between an empirical and theoretical distribution (see panels D and E in Figure 8). We determined the chi-square distribution based on 1000 realizations of the empirical distribution for (A) 20 trials and (B) 100 trials. The data are drawn from (dark bars) the pattern model or (gray bars) the equivalent null hypothesis model and are compared to the corresponding theoretical null hypothesis model. In Panel B, pattern models are perfectly distinguishable from the null hypothesis model because the distributions do not overlap. (C) The distinguishability is quantified using the area under the ROC curve, which varies between 0.5 (not distinguishable) and 1 (perfectly distinguishable).

As  $q_{\max}$  reflects the trade-off between the precision ( $\sigma$ ) and the reliability, it should depend systematically on the value of  $\sigma$ . Across different realizations, the  $dCV_d$  and the entropy of the distance distribution ( $S_d$ ) vary, which means that, put simply, the  $q_{\max}$  estimate is reliable only when the variability in these quantities is less than the size of the landmarks in those curves whose position determines the value of  $q_{\max}$  (see Figure 10A). For this specific event structure, the mean  $q_{\max}$  value reaches a constant value when the number of trials exceeds 10 to 60, depending on the precision (see Figure 10B). The standard deviation across realizations in  $q_{\max}$  decreases with the number of trials for all precisions except for  $\sigma = 5$  ms (see Figure 10C).

The number of clusters  $N_c$  is determined as the location of the peak in the differenced gap statistic  $dG(N_c)$ . We find that the height of this peak at  $N_c = 4$  (the correct value) decreased with  $\sigma$ , being virtually indistinguishable for  $\sigma = 5$  ms (see Figure 10D). The variability in  $dG(N_c)$  (see Figure 10E) decreases with the number of trials, which means that on a large fraction of the realizations, the  $N_c$  is indeed correctly given by the peak in  $dG$  (see Figure 10F). For  $\sigma = 1$  ms, this fraction saturates at 1 (always the correct  $N_c$ ) when  $N_{\text{trial}}$  exceeded 50 (see Figure 10F). The best performance for higher  $\sigma$  was less than 1, and for higher  $\sigma$ , the fraction decreased with  $N_{\text{trial}}$  at higher values of  $N_{\text{trial}}$  (see Figure 10F).

The analysis procedure tries to assign each spike to an event, after which the event statistics—precision, reliability, and mean spike time—can be calculated and the pattern significance determined. There are two ways in which mistakes can be made (see Figure 10G). First, there are too few or too many events. The too-few-events case occurs either because the number of spikes in an event is less than three (see section 4.1.4), which is common when there are not enough trials (these spikes are assigned to be noise), or because events are erroneously merged, which happens when the constituent events have low precision. The too-many-events case is due to splitting up broad, low-precision events. Second, trials are assigned to the wrong cluster, which happens sometimes because of inadequate values of  $q_{\max}$ ,  $N_c$ , or imprecision. This leads to events that are inadvertently split across multiple clusters and subsequently not merged.

To analyze these two contributions, we compared the number of events found using the analysis procedure to the number that was expected (11 in this case; see Figure 10H). When the number of trials exceeded  $N_{\text{trial}} = 50$ , the number of events approached 11. We also determined the normalized mutual information  $I_N$  between the cluster assignment and the correct class. It did not vary systematically with trial number but decreased as the jitter increased (see Figure 10I).

Taken together, this implies that if there are well-defined patterns, 50 to 60 trials are enough to characterize 4 patterns, that is, as a rule of thumb, about 10 trials per pattern. These numbers depend on the precise spike times, reliabilities, and precisions of the underlying patterns and the number of patterns, but the procedure outlined here can be used to determine

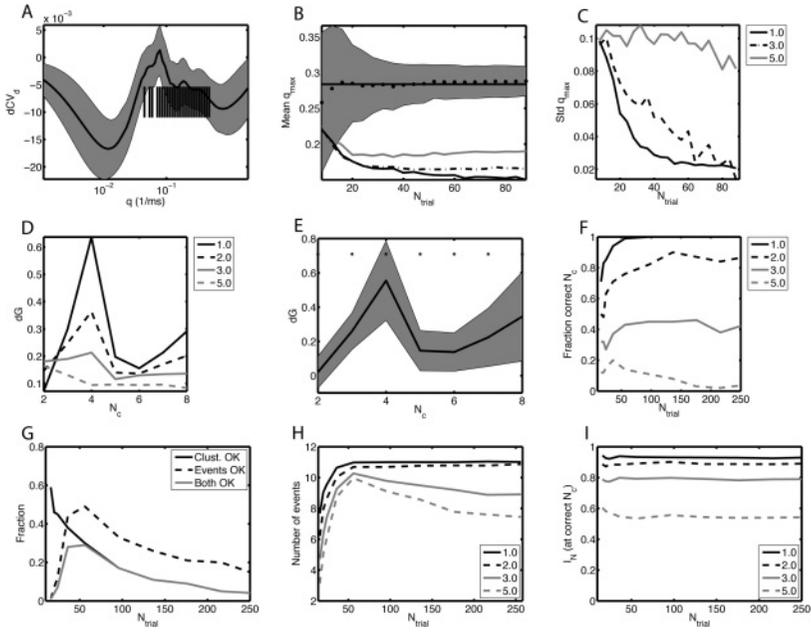


Figure 10: For a 4-pattern, 11-event example data set, 10 trials per pattern proved sufficient for performing the event-finding analysis. (A) The mean of  $dCV_d$  versus  $q$ , with the gray bands indicating plus or minus the standard deviation across 500 realizations (other parameters:  $\sigma = 1$  ms,  $N_{\text{trial}} = 16$ ). Each vertical tick represents the picked  $q$  values for a realization. (B) The mean of the  $q_{\text{max}}$  value picked by the heuristic (see the text) as a function of  $N_{\text{trial}}$  for  $\sigma = 1$  ms. We show the actual value (filled circles), the mean across all the  $N_{\text{trial}}$  values (solid horizontal line), and plus or minus the standard deviation across realizations (gray band). For comparison, the curves are also shown for  $\sigma = 2$  ms (gray solid line), 2.5 ms (black dot-dashed line), and 3 ms (black solid line). (C) The standard deviation in  $q_{\text{max}}$  across 500 realizations versus  $N_{\text{trial}}$  for different  $\sigma$  values as indicated in the legend. (D) The differenced gap statistic  $dG$  versus the number of clusters  $N_c$  for different  $\sigma$  values as indicated in the legend. The actual number of patterns or clusters is four. (E)  $dG$  versus  $N_c$  for  $\sigma = 1$  ms,  $N_{\text{trial}} = 16$ . The gray band indicates plus or minus the standard deviation across 100 realizations. (F) The fraction of realizations for which the  $N_c$  values found using the gap statistic criterion are equal to the actual number of patterns for different  $\sigma$  values as indicated in the legend. (G) The fraction of realizations versus  $N_{\text{trial}}$  for which each trial is assigned to the correct pattern (solid line), the number of detected events is equal to the correct number (11) (dashed line), and for which both these criteria are satisfied (gray solid line) for  $\sigma = 1$  ms. (H) The mean number of events detected and (I)  $I_N$  (the normalized mutual information between the found clustering and the actual pattern assignment) versus  $N_{\text{trial}}$  for different  $\sigma$  values as indicated in the legend.

the number of trials that is adequate for the specific situation under consideration.

**4.2 Finding Spike Patterns and Determining the Event Structure in Experimental Data.** We applied our event-finding procedure to an experimental data set comprising 11 experiments on 10 different cells with between 18 and 51 trials per condition (see Table 1). These data were collected for the purpose of a different analysis, which will be published elsewhere. A current comprising a constant current step (“offset”) and a fluctuating drive was injected in the soma of a layer 5 pyramidal cell. The fluctuating drive was exactly the same on each trial, but we used 11 different amplitudes (the factor by which the fluctuating drive was multiplied before it was added on top of the current step), which were expressed as percentages. Because the injected waveform was prepared offline and stored in a file, at the time of recording we could only adapt the overall gain to the properties of the neuron we recorded from. The overall gain was chosen such that the neuron produced spikes for the lowest value of the amplitude, which was achieved for 9 out of 10 cells.

In Figure 11 we show rastergrams for four example cells, along with the template of the injected current waveform (without the current step and scaling factor), shown at the bottom of each panel. The rastergram consisted of blocks of constant amplitude, with the highest amplitude on top. Within each block, the trials were in the order they were collected, with the earliest at the bottom. In the rastergram of Figure 11A, events are visible as lines of spike alignments that appear for low to intermediate amplitudes and become sharper when the amplitude is increased further. Overall this graph suggests that the precision and the reliability improved with amplitude (see section 3 for a precise definition). The events were associated with large depolarizing fluctuations, for example, the one at 200 ms (see Figure 11C, double-headed arrows)

A second example is shown in Figure 11B. The overall behavior was similar except that the cell showed more adaptation. For zero amplitude, this cell stopped spiking after about 500 ms, whereas the cell in panel A continued spiking to the end of the trial. We characterized this in terms of the ratio of the mean of the second interspike interval (ISI) divided by the mean of the first ISI. This was 0.92 for panel A and 1.1 for panel B. For the example in panel C, new events emerged as the amplitude increased, but the position and precision of the other events appeared not to be affected. The rastergram in panel D shows a response that did not strongly depend on amplitude. However, this example is instructive because it shows the type of nonstationary responses encountered in the data. Consider the time interval centered around 650 ms (indicated by the arrow). A spike was present in this interval on the later trials but not during the earlier trials. In general we attempted to record as many trials as possible, and we stopped only when the cell could not maintain a stable membrane potential or input

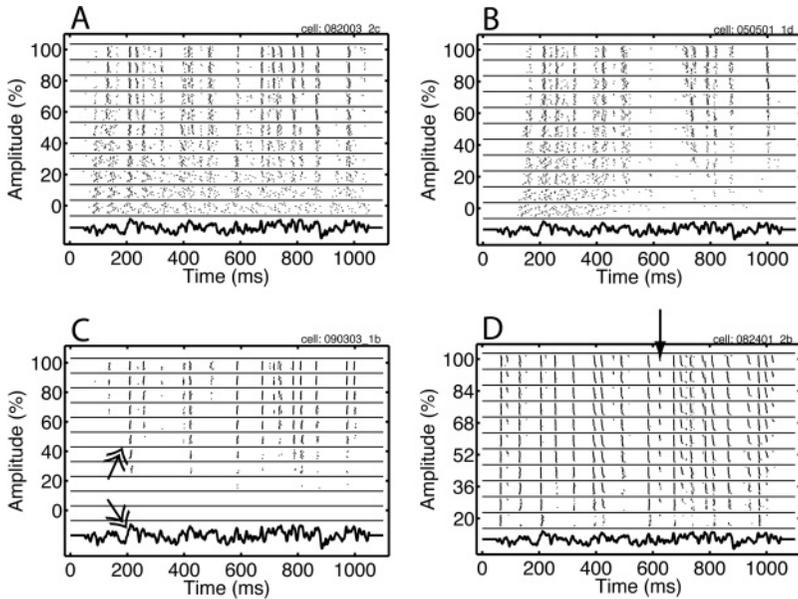
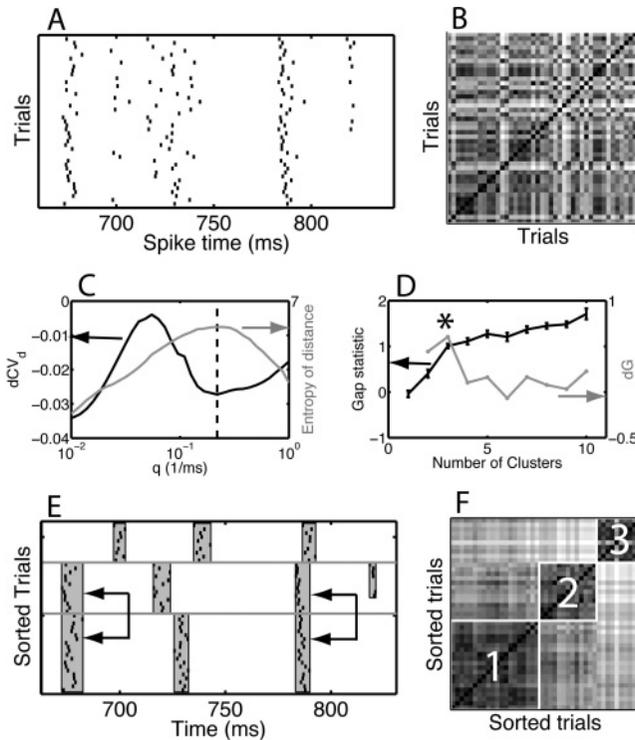


Figure 11: Rastergrams obtained in response to injection of the same fluctuating current waveform for four different neurons. Each line in the rastergram represents a spike train obtained on a trial, with the  $x$ -ordinate of each tick being the spike time. The spike trains are ordered in blocks (delineated by horizontal lines) based on the amplitude of the injected current, expressed as a percentage, with the highest amplitude on top. Within each block, the trials are ordered as they were recorded, with the earliest trial at the bottom. The injected current is shown for a single amplitude at the bottom. Each panel shows the response of an example cell, the details of which are in the main text. The arrow indicates an example of nonstationarity; the double-headed arrow highlights a large depolarizing fluctuation and the corresponding event (see the text for details). The maximum amplitude (including online gain) in nA was 0.08, 0.05, 0.09, and 0.1, and the offset (again including online gain) was 0.12, 0.075, 0.1350, and 0.15, for panels A–D, respectively.

resistance between the stimulation periods. Therefore, the last few trials usually were different and were discarded prior to further analysis. The event-finding algorithm was robust against the effects of nonstationarity.

We illustrate the event-finding procedure using the data shown in Figure 11A at an amplitude of 70% and in the time segment starting at 650 ms and ending at 850 ms (see Figure 12A). Because the events were overlapping they could not be separated using an algorithm based on an ISI threshold. First, the similarity matrix was determined (see Figure 12B). The pixels in the  $i$ th and  $j$ th rows represent the VP distance between trial  $i$  and  $j$  (Victor

& Purpura, 1996). The VP metric requires a temporal resolution parameter  $q$ , which represents a trade-off between reliability and precision: a pair of spikes between the two spike trains was considered different when their difference exceeded  $2/q$ . The  $q$  should be chosen such that spike patterns were most distinguishable. Our heuristic was based on how the distribution of distance values (the elements in the distance matrix) varied as a function of  $q$ . We used the entropy obtained by binning the distribution in 200 bins (see Figure 12C, gray line) and the logarithmic derivative of the coefficient of variation of the distances ( $CV_d$ ). The  $CV_d$  is the standard deviation of the distance divided by its mean. The log derivative was approximated by differencing the  $CV$  at logarithmically spaced  $q$  values and is referred to as the  $dCV_d$  (see Figure 12C, black line). For small  $q$ , the distance between two spike trains was equal to the difference in spike counts, which means it can take only close-to-integer values yielding a low value for the entropy. For large  $q$ , each spike in the two spike trains was considered different. Hence, the distance was again close to integer and of low entropy because it was the sum of spike counts. For intermediate  $q$  values, there was a broad peak, for which spike patterns were distinguishable. The  $dCV_d$  was more sensitive to the specific  $q$  value as indicated by the peak and troughs, each of which



reflected a change in the spike pattern feature the measure was sensitive to. We found that the most robust estimate was produced by taking the mean of these two values (see the dashed vertical line in Figure 12C).

The second step was to cluster the distance matrix at the chosen  $q$  value. Each column was interpreted as a vector in a high-dimensional space. For computational efficiency, the dimensionality was reduced by applying principal component analysis and retaining only the 10 components with the highest variance. (We found 10 components to be adequate for our data set, but for different data sets, a procedure would be necessary to pick the number of components based on the desired fraction of the data variance to be retained in the components, Jolliffe, 2002.) These data were then analyzed

---

Figure 12: The procedure for detecting and characterizing events. (A) Rastergram of the data in Figure 11A for an amplitude of 70% and the time segment between 650 ms and 850 ms. (B) Distance matrix for the trials shown in panel A. The pixel on the  $i$ th row and  $j$ th column represents the Victor-Purpura distance between the spike train on trial  $i$  and the one on trial  $j$  using the selected value for the temporal resolution parameter  $q$ . The range from small to large distances is mapped onto a gray scale going from dark to white. (C) The heuristic for determining  $q$  is based on the distribution of the elements of the distance matrix, referred to as the distances. We show (black curve, left-hand-side scale) the  $dCV_d$  and (gray curve, right-hand-side scale) the entropy of the distances as a function of  $q$ . The coefficient of variation was calculated as the ratio of the standard deviation of the distances over their mean at  $q$ -values whose  $\log_{10}$  values were uniformly distributed. The  $dCV_d$  is the difference between consecutive  $CV_d$  values and thus corresponds to a logarithmic derivative. The entropy is obtained by binning the distances and using the standard  $p \log p$  expression (see section 3.4). The  $q$  value chosen by the heuristic is the mean of the  $q$  value at which the entropy has a maximum and the location of the deepest trough in the  $dCV_d$  that occurs after the highest peak (dashed vertical line). (D) The heuristic for determining the number of spike patterns. The FCM algorithm is used to find  $N_c$  clusters in the data, after which the gap statistic  $G(N_c)$  is computed. Each cluster is hypothesized to be a spike pattern. The gap statistic measures the reduction of within-cluster variance achieved by clustering relative to a similarly clustered surrogate data set with points uniformly distributed in the hypercube spanned by the range of the original data. The discrete derivative of  $G$  is  $dG(N_c) = G(N_c) - G(N_c - 1)$ . We show the (black curve, left-hand-side scale)  $G(N_c)$  and (gray curve, right-hand-side scale)  $dG(N_c)$ . The error bars are the standard deviation obtained across 50 surrogates. The value of  $N_c$  chosen by the heuristic is the one for which  $dG$  is maximal (asterisk). (E) Rastergram and (F) distance matrices with the trials reordered according to their cluster membership. The horizontal lines in panel E separate the clusters, and the vertical gray bands are the events. On two instances, events common to more than one cluster were merged, as indicated by the arrows.

using the FCM procedure with the “fuzzifier” parameter kept at its default value of 2. This procedure required the number of clusters to be specified, which we determined using the differenced gap-statistic  $dG$  (Tibshirani et al., 2001), which is a function of the number of clusters. The gap statistic measures the improvement of the within-cluster variance relative to data, clustered by the same method, in which there are no clusters because the data points fill the space uniformly within the range of the original data. When used on data for which the gap statistic was developed, the gap statistic had a peak at the actual number of clusters. For our data, the gap statistic initially increased quickly with the number of clusters, followed by a smaller rate of increase. We found that the number of clusters reached after the largest increase, which is indicated by the peak in the differenced gap statistic (see the asterisk in Figure 12D), in most cases led to the same number of clusters that would be selected manually. This heuristic identified three clusters in the data (see Figure 12F). In Figure 12E, the clusters (spike patterns) are separated by horizontal lines. Within each cluster, the events were well separated so they could easily be found by the ISI-based method (Tiesinga et al., 2002).

Across clusters, there were events with similar spike times, which needed to be merged (example: first spike time in cluster 1 and 2 in Figure 12E). An ROC analysis was used to determine how distinguishable the two groups of spikes were. This is a standard procedure to evaluate the results of the two-alternative forced-choice task (Green & Swets, 1966). The ROC was rescaled to yield the sROC (see section 4.1.5), which takes values between 0 and 1. If the two groups are non overlapping, the sROC was equal to 1. The merging procedure was characterized by a threshold for sROC below which groups of events were considered the same. The merged events in Figure 12E, indicated by overlapping gray background and the arrows, were obtained using a threshold of 0.5.

The events visible in the multi-amplitude rastergrams (see Figure 11) persisted across amplitudes. This meant that the precision, reliability, and mean spike time of events could be compared across amplitudes. One strategy is to use the procedure outlined in Figure 12 to find events for each amplitude separately, after which common events across amplitudes are merged. In Figure 13 we show the results of an alternative strategy, where the five highest amplitudes are analyzed at the same time. This approach was superior to separate analyses because the analysis procedure worked better when there were more trials in each cluster (see section 4.1.7). Our analysis revealed the presence of four clusters (see Figures 13C and 13D), which led to eight events, some of which were common to multiple patterns. The persistence of patterns across multiple amplitudes is a common phenomenon in our experimental data and can be explained using the mathematical concept of spike train bifurcations: spike times vary smoothly with amplitude until a bifurcation point is reached and the spike times change abruptly; it is even discontinuous in some model neurons. The relation between the

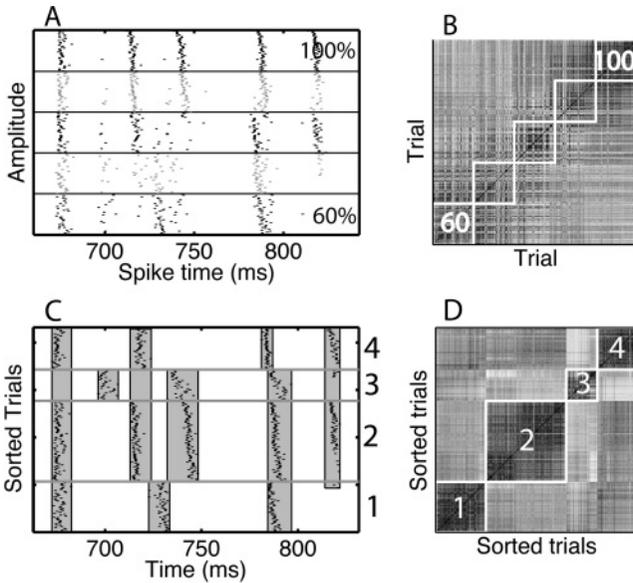


Figure 13: Spike patterns persisted across multiple amplitudes. (A) The rastergram for the data shown in Figure 11A for amplitudes between 60% and 100% and during the time segment between 650 ms and 850 ms. (B) The corresponding distance matrix obtained for the  $q$  values selected by the heuristic. (C, D) The gap statistic suggested four patterns. We show the (C) rastergram and (D) distance matrix with the trials sorted according to their cluster membership. The numbers indicate the cluster index. The gray vertical bands show the eight detected events that remained after applying the merging procedure.

experimental observations and the underlying mathematical phenomena will be the subject of a future publication.

## 5 Discussion

---

We developed a four-step procedure with a small number of parameters to determine the event structure of a set of spike trains (see Figure 2). For this procedure to be applicable, the histogram needs to be peaked, which is indicative of events. It could also deal with overlapping peaks as long as they were due to multiple spike patterns. The procedure was tested using artificial data with embedded spike patterns. It was also applied to spike trains recorded in response to the same fluctuating current injected across multiple trials. The method uncovered evidence for spike patterns in these data, the relevance of which is discussed below. The procedure can also be applied to spike trains from groups of neurons, for instance,

a set of inhibitory and excitatory neurons. It can then be used to separate the inhibitory and excitatory responses and determine the precision and relative lag, which are objects of experimental and theoretical interest (Buia & Tiesinga, 2006; Mishra, Fellous, & Sejnowski, 2006; Womelsdorf et al., 2007).

Our results provide evidence for within-trial correlations (spike patterns) and thereby confirm previous results obtained in various experimental preparations. For instance, reverse correlation experiments have been conducted to determine which stimulus features generate spikes. For experiments where the stimulus was a current injected at the soma, the spike-triggered average stimulus waveform usually consisted of a hyperpolarization followed by a depolarization. The depth and duration of the hyperpolarization depended on when the preceding spike occurred (Jolivet et al., 2004; Powers, Dai, Bell, Percival, & Binder, 2005), which means that the reverse correlation reflects both the stimulus features and the spiking history of the neuron. These effects were well captured using integrate-and-fire (Pillow et al., 2005) or spike-response model neurons (Jolivet et al., 2004, 2006). When the spike history and adaptation effects were incorporated explicitly, these models could account for the precision and reliability of *in vitro* neurons driven by current injection at the soma. In addition, with the proper stimulus filter, these spiking models could also account for the responses of retinal ganglion cells driven by visual stimuli (Keat et al., 2001; Pillow et al., 2005). The data from these experiments show evidence for the same type of spike patterns analyzed here (see Figure 9 in Pillow et al., 2005) and in our previous work (Fellous et al., 2004; Tiesinga & Toups, 2005).

Although the type of dynamics examined here can be captured in terms of relatively simple models, the analysis method we present has a number of advantages. It does not depend on a fitting procedure, and it does not require knowledge of the explicit stimulus waveform other than its starting time. The analysis method requires fewer data than would be necessary to fit a model. This is an advantage for *in vivo* data obtained from awake, behaving subjects, where the cell is held for only a limited amount of time and there is no direct control of the current drive at the spike generator. *In vivo* data also show that the same neuron can change its spiking dynamics from regular spiking to intrinsically bursting (Steriade, 2004), which is significant because it had been thought that these two spiking dynamics represent two different types of neurons (McCormick, Connors, Lighthall, & Prince, 1985). A consequence of this result is that the parameters of the fit will change depending on cortical state, which is where the power of the clustering method becomes clear. The method described here should be able to detect the difference in dynamics and place the spike trains in different clusters.

As observed in other recent studies, we found that neurons can become synchronized to a common input (de la Rocha, Doiron, Shea-Brown, Josic, & Reyes, 2007; Ermentrout et al., 2008; Markowitz, Collman, Brody,

Hopfield, & Tank, 2008; Tiesinga et al., 2008; Tiesinga & Troups, 2005). These neurons form an ensemble, whose collective postsynaptic impact is enhanced because of synchrony (Salinas & Sejnowski, 2000, 2001; Wang, Spencer, Fellous, & Sejnowski, 2010). In our data, the most precise and reliable responses were obtained when the neuron produced one pattern, which was at the highest value for the amplitude. Hence, this interpretation suggests that the single pattern state is best from the viewpoint of information transmission. However, what state provides the most information about the temporal fluctuations in the stimulus waveform? Reverse correlation studies show that spikes are most often elicited in response to a transient depolarization (Powers et al., 2005), that is, to peaks in the stimulus waveform. At high-amplitude values, the spike rate, which is determined by the afterhyperpolarization currents, may cause the neuron to produce a spike on only a fraction (say, 50%) of the stimulus peaks. Under these circumstances, information about the stimulus waveform could be lost. However, when there are two spike patterns, each of which spikes on half of these peaks, all the peaks are represented in the ensemble discharge. In a companion paper (Troups, Fellous, Thomas, Sejnowski, & Tiesinga, 2011), we show using this analysis that this situation can lead to better stimulus representation at intermediate amplitudes because the event precision remains high. The multiple spike pattern case thus represents an optimal compromise with a relatively high precision for information transmission and good coverage of the temporal features of the stimulus waveform.

We plan to use the event-finding method in a future study to determine how the precision and reliability, calculated as independent variables, vary with the amplitude of the injected current waveform. Within the context of this goal, we found that the procedure was less efficient for the lower-amplitude data sets, where the precision was reduced and the response was more sensitive to drifts. This observation is consistent with predictions based on the effect of reduced precision in the simulated data sets.

The Matlab code for this analysis together with example data sets will be made available on <https://github.com/VincentTroups/matlab-utils/>.

## Acknowledgments

---

This research was supported in part by the National Institutes of Health (R01-MH68481); start-up funds from the University of North Carolina at Chapel Hill and the Human Frontier Science Program (J.V.T. and P.H.T.); the National Science Foundation (DMS-0720142) (P.J.T.); and the Howard Hughes Medical Institute (T.J.S.). P.J.T. acknowledges research support from the Oberlin College Library.

## References

---

- Bair, W., & Koch, C. (1996). Temporal precision of spike trains in extrastriate cortex of the behaving macaque monkey. *Neural Comput.*, *8*, 1185–1202.
- Bezdek, J. C. (1981). *Pattern recognition with fuzzy objective function algorithms*. New York: Plenum.
- Bouguessa, M., Wang, S. R., & Sun, H. J. (2006). An objective approach to cluster validation. *Pattern Recognition Letters*, *27*, 1419–1430.
- Buia, C., & Tiesinga, P. (2006). Attentional modulation of firing rate and synchrony in a model cortical network. *Journal of Computational Neuroscience*, *20*, 247–264.
- Buracas, G. T., Zador, A. M., DeWeese, M. R., & Albright, T. D. (1998). Efficient discrimination of temporal patterns by motion-sensitive neurons in primate visual cortex. *Neuron*, *20*, 959–969.
- Cover, T. M., & Thomas, J. A. (1991). *Elements of information theory*. Hoboken, NJ: Wiley.
- de la Rocha, J., Doiron, B., Shea-Brown, E., Josic, K., & Reyes, A. (2007). Correlation between neural spike trains increases with firing rate. *Nature*, *448*, 802–806.
- Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification* (2nd ed.). Hoboken, NJ: Wiley.
- Ermentrout, G. B., Galan, R. F., & Urban, N. N. (2008). Reliability, synchrony and noise. *Trends Neurosci.*, *31*, 428–434.
- Fellous, J. M., Houweling, A. R., Modi, R. H., Rao, R. P., Tiesinga, P. H., & Sejnowski, T. J. (2001). Frequency dependence of spike timing reliability in cortical pyramidal cells and interneurons. *J. Neurophysiol.*, *85*, 1782–1787.
- Fellous, J. M., Tiesinga, P. H., Thomas, P. J., & Sejnowski, T. J. (2004). Discovering spike patterns in neuronal responses. *J. Neurosci.*, *24*, 2989–3001.
- Green, D. M., & Swets, J. A. (1966). *Signal detection theory and psychophysics*. Hoboken, NJ: Wiley.
- Haider, B., & McCormick, D. A. (2009). Rapid neocortical dynamics: Cellular and network mechanisms. *Neuron*, *62*, 171–189.
- Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol.*, *160*, 106–154.
- Jolivet, R., Lewis, T. J., & Gerstner, W. (2004). Generalized integrate-and-fire models of neuronal activity approximate spike trains of a detailed model to a high degree of accuracy. *J. Neurophysiol.*, *92*, 959–976.
- Jolivet, R., Rauch, A., Luscher, H. R., & Gerstner, W. (2006). Predicting spike timing of neocortical pyramidal neurons by simple threshold models. *J. Comput. Neurosci.*, *21*, 35–49.
- Jolliffe, I. T. (2002). *Principal component analysis* (2nd ed.). New York: Springer.
- Keat, J., Reinagel, P., Reid, R. C., & Meister, M. (2001). Predicting every spike: A model for the responses of visual neurons. *Neuron*, *30*, 803–817.
- Larsen, R. J., & Marx, M. L. (1986). *An introduction to mathematical statistics and its applications* (2nd ed.). Upper Saddle River, NJ: Prentice Hall.
- Mainen, Z. F., & Sejnowski, T. J. (1995). Reliability of spike timing in neocortical neurons. *Science*, *268*, 1503–1506.

- Markowitz, D. A., Collman, F., Brody, C. D., Hopfield, J. J., & Tank, D. W. (2008). Rate-specific synchrony: Using noisy oscillations to detect equally active neurons. *Proc. Natl. Acad. Sci. USA*, *105*, 8422–8427.
- McCormick, D. A., Connors, B. W., Lighthall, J. W., & Prince, D. A. (1985). Comparative electrophysiology of pyramidal and sparsely spiny stellate neurons of the neocortex. *J. Neurophysiol.*, *54*, 782–806.
- Mishra, J., Fellous, J. M., & Sejnowski, T. J. (2006). Selective attention through phase relationship of excitatory and inhibitory input synchrony in a model cortical neuron. *Neural Netw.*, *19*, 1329–1346.
- Pal, N. R., & Bezdek, J. C. (1995). On cluster validity for the fuzzy C-means model. *IEEE Transactions on Fuzzy Systems*, *3*, 370–379.
- Panzeri, S., & Treves, A. (1995). Analytical estimates of limited sampling biases in different information measures. *Network: Comput. Neural Syst.*, *7*, 87–107.
- Pillow, J. W., Paninski, L., Uzzell, V. J., Simoncelli, E. P., & Chichilnisky, E. J. (2005). Prediction and decoding of retinal ganglion cell responses with a probabilistic spiking model. *J. Neurosci.*, *25*, 11003–11013.
- Powers, R. K., Dai, Y., Bell, B. M., Percival, D. B., & Binder, M. D. (2005). Contributions of the input signal and prior activation history to the discharge behaviour of rat motoneurons. *J. Physiol.*, *562*, 707–724.
- Reinagel, P., Godwin, D., Sherman, S. M., & Koch, C. (1999). Encoding of visual information by LGN bursts. *J. Neurophysiol.*, *81*, 2558–2569.
- Reinagel, P., & Reid, R. C. (2000). Temporal coding of visual information in the thalamus. *J. Neurosci.*, *20*, 5392–5400.
- Reinagel, P., & Reid, R. C. (2002). Precise firing events are conserved across neurons. *J. Neurosci.*, *22*, 6837–6841.
- Rezaee, M. R., Lelieveldt, B. P. F., & Reiber, J. H. C. (1998). A new cluster validity index for the fuzzy C-mean. *Pattern Recognition Letters*, *19*, 237–246.
- Rieke, F., Warland, D., de Ruyter van Steveninck, R. R., & Bialek, W. (1997). *Spikes: Exploring the neural code*. Cambridge, MA: MIT Press.
- Rohatgi, V. K. (2003). *Statistical inference*. Mineola, NY: Dover.
- Salinas, E., & Sejnowski, T. (2000). Impact of correlated synaptic input on output variability in simple neuronal models. *J. Neurosci.*, *20*, 6193–6209.
- Salinas, E., & Sejnowski, T. J. (2001). Correlated neuronal activity and the flow of neural information. *Nature Reviews Neuroscience*, *2*, 539–550.
- Steriade, M. (2004). Neocortical cell classes are flexible entities. *Nat. Rev. Neurosci.*, *5*, 121–134.
- Strong, S., Koberle, R., de Ruyter van Steveninck, R., & Bialek, W. (1998). Entropy and information in neural spike trains. *Phys. Rev. Letts.*, *80*, 197–200.
- Tibshirani, R., Walther, G., & Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society Series B, Statistical Methodology*, *63*, 411–423.
- Tiesinga, P. H., Fellous, J. M., & Sejnowski, T. J. (2002). Attractor reliability reveals deterministic structure in neuronal spike trains. *Neural Comput.*, *14*, 1629–1650.
- Tiesinga, P., Fellous, J. M., & Sejnowski, T. J. (2008). Regulation of spike timing in visual cortical circuits. *Nat. Rev. Neurosci.*, *9*, 97–107.
- Tiesinga, P. H. E., & Toups, J. V. (2005). The possible role of spike patterns in cortical information processing. *Journal of Computational Neuroscience*, *18*, 275–286.

- Toups, J. V., Fellous, J.-M., Thomas, P. J., Sejnowski, T. J., & Tiesinga, P. H. (2011). *Multiple spike train patterns occur at bifurcation points of membrane potential dynamics*. Unpublished manuscript.
- Victor, J. D., & Purpura, K. P. (1996). Nature and precision of temporal coding in visual cortex: A metric-space analysis. *J. Neurophysiol.*, *76*, 1310–1326.
- Wang, H. P., Spencer, D., Fellous, J. M., & Sejnowski, T. J. (2010). Synchrony of thalamocortical inputs maximizes cortical reliability. *Science*, *328*, 106–109.
- Womelsdorf, T., Schoffelen, J. M., Oostenveld, R., Singer, W., Desimone, R., Engel, A. K., et al. (2007). Modulation of neuronal interactions through neuronal synchronization. *Science*, *316*, 1609–1612.
- Yan, M., & Ye, K. (2007). Determining the number of clusters using the weighted gap statistic. *Biometrics*, *63*, 1031–1037.
- Zahid, N., Limouri, N., & Essaid, A. (1999). A new cluster-validity for fuzzy clustering. *Pattern Recognition*, *32*, 1089–1097.

---

Received November 5, 2010; accepted February 7, 2011.