8

Analysis of Extracellular Recordings

8.1. Introduction

This chapter presents the methods of analysis required for the “oldest” types of brain–machine interface. These methods are (strongly) invasive, as they require trepanation to insert a large number (10–100) of electrodes into the brain tissue. The first feasibility studies were performed on rats [CHA 99] and on monkeys [WES 00] – Chaplin [CHA 04] presents an initial overview of this work. The advantage of these methods of recording is that they give access to individual neuron activity with excellent resolution in time (achieving this resolution is the main subject of this chapter) and the obvious disadvantage is that they require trepanation, which excludes them from being used with patients, except in very exceptional cases. As the other chapters of this book will discuss, BCIs may be implemented without recourse to the invasive methods that we shall discuss here; however, these methods are still very frequently used by neurophysiologists in a wider context that we shall introduce in the following section.

Chapter written by Christophe POUZAT.

Brain–Computer Interfaces 1: *Foundations and Methods*, First Edition.
Edited by Maureen Clerc, Laurent Bougrain and Fabien Lotte.
© ISTE Ltd 2016. Published by ISTE Ltd and John Wiley & Sons, Inc.

8.1.1. *Why is recording neuronal populations desirable?*

There are three main reasons why recording large numbers of neurons simultaneously while maintaining the resolution of individual neurons is desirable for neurophysiologists[1]:

1) more data collected per instance of the experiment, which limits the number of animals required for a study and reduces costs;

2) multiple models of information processing by neuronal networks, such as perceptual binding by synchronization[2], suggest that the synchronization of the activity of certain neurons plays a defining role [MAL 81], and the simultaneous recording of multiple neurons strongly facilitates or is perhaps even necessary for the experimental study of this kind of model [DON 08];

3) multiple examples such as that of the motor system [GEO 86] show that, even without synchronization, *groups of neurons* are required to properly represent a stimulus or an action such as a motor command; even though it is sometimes possible to study these phenomena through successive recordings of unique neurons (as was performed in [GEO 86]), simultaneous recordings make this task a lot easier (which returns to the first point outlined above).

8.1.2. *How can neuronal populations be recorded?*

There are currently three methods for recording neuronal populations. Multiple extracellular recordings [BUZ 04] are the most commonly used technique. The subject of this chapter is the analysis of data obtained using this technique. Recordings with potential-sensitive probes [ZEC 89, HOM 09] are used on brain slices and ganglia of invertebrates. Finally, recordings using calcium-based fluorescence [HOM 09] are often presented in the literature, but their resolution in time is insufficient for the study of questions of synchronization (see Figure 4 in Chapter 4 of [CAN 10]).

[1] The technique of EEG, extensively discussed throughout this book, also provides simultaneous recordings of multiple neurons, but without the resolution of individual neurons.

[2] See: http://en.wikipedia.org/wiki/Binding_problem.

8.1.3. *The properties of extracellular data and the necessity of spike sorting*

Figure 8.1 shows 1 s of recording at the four sites of a *tetrode* [GRA 95]. This recording was performed in the first olfactory relay, the *antennal lobe*, of an insect: the locust *Schistocerca americana*. These readings will serve as a running example throughout this chapter. Before being converted into a numerical format (at a sample rate of 15 kHz), the data were filtered between 300 Hz and 5 kHz; the full details of the recordings are given in [POU 02]. The reader should note that the 300 Hz high-pass filter will have removed most of the *local field potentials* that arise from postsynaptic activity. To keep this chapter brief, we will not discuss the analysis of this type of signal; they are identical to signals obtained by intracranial EEG[3].

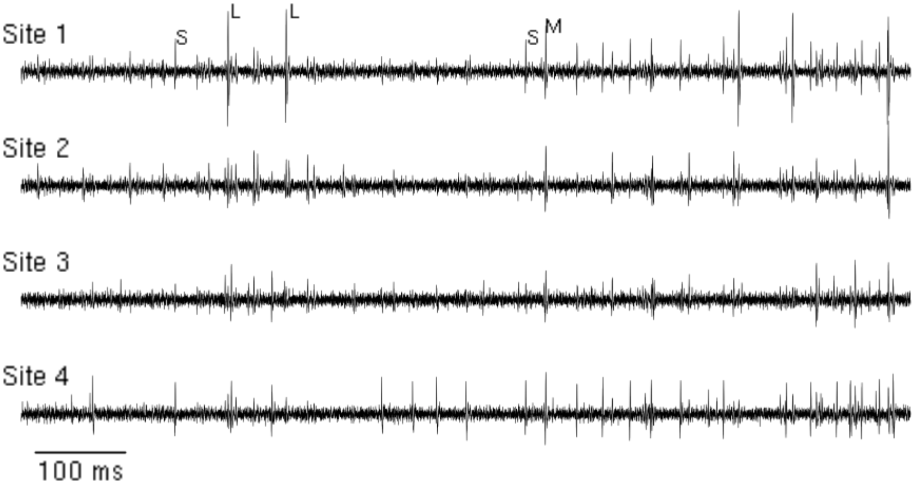

Figure 8.1. *One second of data recorded at the four sites (electrodes) of a tetrode. The data were filtered between 300 Hz and 5 kHz before being converted into a numerical format. The sample rate is 15 kHz. These readings were taken in the antennal lobe of a locust* Schistocerca americana*. Examples of action potentials at the first site are marked by the letters S, M and L, with small, medium, and large amplitudes, respectively*

[3] See: https://en.wikipedia.org/wiki/Electrocorticography.

The spikes visible at each of the sites of the recording, some examples of which have been marked by the letters S (small), M (medium) and L (large) in Figure 8.1, are of particular interest to us. These spikes are generated by action potentials emitted by multiple neurons [PLO 07]. Based on the "all-or-nothing" property of action potentials *in axons*, we can conclude, at least provisionally, that multiple neurons are present [ADR 22]. Given these type of data, we start by asking the following two interrelated questions:

1) How many neurons contributed to the recording?

2) Which neuron was the originator of each of the visible spikes?

Spike sorting is the stage of data processing that attempts to answer these questions; we will study it in this chapter. We will return to the question of the origin of the signal and a justification of the use of a tetrode, which consists of multiple electrodes in close proximity of each other. The reader should, however, note at this point that the spikes marked with S appear to be associated with signals of similar amplitude on the fourth site, even smaller amplitude on the third site and zero apparent amplitude on the second site. In contrast, the amplitude of the signals associated with spikes marked with M (medium) appears to be constant across all sites. We will see that the distance between the source of the current, i.e. the neuron, and the electrode is the principal determining factor of the recorded signal; given a fixed source (neuron), the *amplitude ratios* are thus functions of the distance ratios between the source and the recording sites: *they depend on the position of the source*. Therefore, different ratios correspond to different sources (neurons).

8.2. The origin of the signal and its consequences

8.2.1. *Relationship between current and potential in a homogeneous medium*

The equation that governs the relationship between the current emitted by a point source[4] of intensity I_0 (in amperes) and electrostatic potential Φ_e (in

[4] Point sources of current are conceptual constructs, in which they does not exist physically, but provide a relevant approximation when a "neurite element" such as an axon segment measuring 0.5 $\mu\mathrm{m}$ or even a soma measuring 15 $\mu\mathrm{m}$ is recorded by an electrode placed at a distance of 100 $\mu\mathrm{m}$.

volts) observed at a distance r (assuming the potential at infinity is zero) is given by:

$$\Phi_e = \frac{1}{4\pi\sigma_e}\frac{I_0}{r}\,, \tag{8.1}$$

where σ_e is the conductivity of the extracellular medium (in Siemens per meter) assumed to be uniform. We shall neglect certain capacitive properties of the extracellular medium [BED 04], but the model developed here, nonetheless, provides an excellent initial approximation [LIN 14]. It is clear that in order to apply equation [8.1], we must first have some method of measuring or estimating the current. The most common approach is to assume a realistic neuron morphology by building in various conductances distributed in a non-homogeneous manner in the membrane, and then numerically solving the cable equation. Solving this equation yields the densities of the various currents. Their sum $i_m(x)$ (where x denotes the lengthwise position within the cable) is then used in a differential version of equation [8.1] to obtain a formula of the following type:

$$\Phi_e = \frac{1}{4\pi\sigma_e}\int_N \frac{i_m(x)}{r(x)}dx\,, \tag{8.2}$$

where the integral is taken along the skeleton of the neuron[5], referred to by the label N. If the value of the membrane potential is known “at each point” along the neuron, then by an elementary application of Ohm’s law and the law of conservation of charge the desired current density may also be obtained. Warning to the reader – the next section requires some mental gymnastics. The membrane potential is usually represented at a fixed position as a function of time, for example in the presence of an action potential. In the next section, we shall fix the time and vary the position. Note that in the case of an action potential propagating at constant speed without deformation, the second representation may be easily deduced from the first.

[5] By skeleton, we mean the true morphology of the neuron after reducing the diameter of each neurite to 0; it is effectively a one-dimensional object with branches embedded in three-dimensional space.

8.2.2. *Relationship between the derivatives of the membrane potential and the transmembrane current*

As described by Rall [RAL 77, pp. 64–65] and Plonsey and Barr [PLO 07, Chapter 8], let us consider a "small segment of neurite" of radius a and length Δx. If the intracellular potential[6] $\Phi_i(x,t)$ at one of the ends of the segment is not equal to the potential $\Phi_i(x+\Delta x,t)$, then Ohm's law states that there is an axial current of intensity:

$$\begin{aligned} I_i(x,t) &= -\pi a^2 \sigma_i \frac{\Phi_i(x+\Delta x,t) - \Phi_i(x,t)}{\Delta x} \\ &\approx -\pi a^2 \sigma_i \frac{\partial \Phi_i(x,t)}{\partial x}\,, \end{aligned} \qquad [8.3]$$

where σ_i is the intracellular conductivity, and where currents are taken to be positive in the direction of increasing x. Now, if the current $I_i(x,t)$ entering the segment is not equal to the current $I_i(x+\Delta x,t)$ exiting the segment, the law of conservation of charge states that the difference must have passed through the membrane; since the density of the transmembrane current $i_m(x,t)$ is positive for outward-flowing current, we obtain:

$$\begin{aligned} I_i(x+\Delta x,t) - I_i(x,t) = -\Delta x\, i_m(x,t) \quad \text{giving} \\ \frac{\partial I_i(x,t)}{\partial x} = -i_m(x,t)\,, \end{aligned} \qquad [8.4]$$

from which we obtain, after combining with equation [8.4]:

$$i_m(x,t) = \pi a^2 \sigma_i \frac{\partial^2 \Phi_i(x,t)}{\partial x^2}\,. \qquad [8.5]$$

Given that the gradients of the observed potentials immediately outside the membrane are much lower than those inside (because the resistance between two external points is much lower than the resistance between two interior points), equation [8.5] becomes:

$$i_m(x,t) = \pi a^2 \sigma_i \frac{\partial^2 V_m(x,t)}{\partial x^2}\,.$$

[6] We explicitly include time, even though initially time is assumed to be fixed.

where V_m is the transmembrane potential, and equation [8.2] may be rewritten as:

$$\Phi_e = \frac{a^2 \sigma_i}{4\sigma_e} \int_L \frac{1}{r(x)} \, \frac{\partial^2 V_m(x,t)}{\partial x^2} \, dx \, . \qquad [8.6]$$

Let us now consider the case of two long neurites with identical properties[7] except for their radius, both capable of propagating an action potential. In the fourth volume of their monumental series, Hodgkin and Huxley solved the wave equation (equation 30 in [HOD 52]). that is to say the ordinary differential equation satisfied by an action potential propagating at constant speed[8]. They also showed that the time scale of the membrane potential does not depend on the radius of the axon, and the speed of propagation θ of the action potential satisfies:

$$\frac{\theta^2}{a \, \sigma_i} = K \, , \qquad [8.7]$$

where K is a constant. Using dimensional analysis, Goldstein and Rall [GOL 74] also showed that the size of the action potential in space is proportional to the square root of the radius – it grows at the same rate as the speed – which implies that, considered as a function of time at any given point of the axon, the action potential does not depend on the radius. These results only hold for *non-myelinated* fibers. The effect of the radius on the spatial profile of the action potential and on its second derivative is shown in the upper section in Figure 8.2, which considers axons of radius 1 μm (left) and 2 μm (right)[9]. We can clearly see that the spatial breadth increases with the diameter, and that the second derivative (necessarily) decreases twice as rapidly with the diameter. The term $1/r(x)$ from equation [8.6] is shown in black (for an electrode situated 50 μm from the center of the axon). Since an

[7] That is, with identical conductances (types and values) and identical plasmic resistivity.

[8] As a historical aside, Hodgkin and Huxley did not solve the system of equations that now bears their names, which involves an equation with partial derivatives, but they did solve – with a mechanical calculator – the simpler system satisfied by a wave propagating without deformation along an axon, i.e. an action potential.

[9] This results come from numerical solutions of the equations of Hodgkin and Huxley obtained using the "classical" parameters specified by them (Detorakis and Pouzat, manuscript in preparation).

analytical solution of equation [8.6] is not possible, numerical solutions for axons with radii between 10^{-1} and 20 μm – the domain of observed values in non-myelinated cortical fibers – are summarized in the graphs at the bottom of the figure. The bottom-left graph of Figure 8.2 shows that the extracellular potential grows faster than the radius to power 1.8; the right-hand graph shows that the extracellular potentially decreases independently of the radius at least as rapidly as one over the square of the distance between the electrode and the axon.

Thus, if an axon of diameter 0.5 $\mu\mathrm{m}$ is connected to a soma of diameter 15 $\mu\mathrm{m}$, the extracellular signal will be dominated by whatever happens inside the soma (Figure 16 in [FAT 57]); otherwise, the axon can be ignored without affecting the value of the extracellular potential. The action potentials recorded by extracellular electrodes will therefore reflect the events that unfold inside the soma and the apical dendrite, if active. This also explains why it is considerably easier to record pyramidal cells (large neurons) than interneurons (small neurons) [GRO 70]. Additionally, if the action potentials of the soma are not identical to those of the axon, as was demonstrated to be the case in experiments[10] for both invertebrates (Figure 13 in [EYZ 55]) and vertebrates (Figure 5A in [WIL 99]), then the relationship between the action potentials recorded by extracellular electrodes and the effective emissions of the neuron is probably not uniquely characterized – the relation: `one somatic action potential = one action potential in the axon` probably does not always hold. We must also consider that propagation may fail at the branching points of the axon (Figure 7 in [ANT 00]) and the possibility of “reflection” of action potentials (Figure 5 in [ANT 00]); two phenomena that may only be properly accounted for with recordings “at all points” in the axon[11]. The conclusion of this short section is that critical thinking remains valuable in the analysis of sequences of action potentials obtained by extracellular recordings (and indeed by intracellular recordings) of the somatic system.

[10] These experiments show that it is possible to have a very small action potential at a somatic level – these somatic recordings are intracellular, which means that these action potentials are very likely indistinguishable from noise in the context of extracellular recordings – together with a perfectly typical action potential in the axon during high-frequency discharges, or *bursts*.

[11] Recordings may be obtained for one unique neuron with membrane potential-sensitive probes.

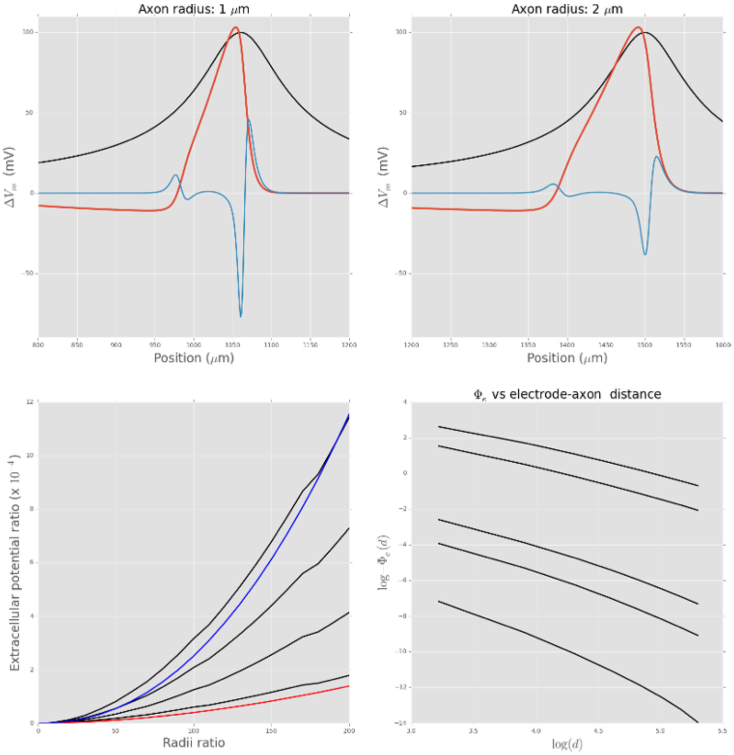

Figure 8.2. *Top: the membrane potential (in red) – expressed as the deviation relative to the rest value – of an action potential of two axons differing only in their radius, 1 μm on the left and 2 μm on the right. In blue, the second derivatives of V_m with respect to the position; the peak value on the left is 0.46 $\mathrm{mV}/\mu\mathrm{m}^2$, and is equal to half of this on the right. In black, the curve shows the term $1/r(x)$ from equation [8.6] for an electrode situated at 50 $\mu\mathrm{m}$ from the center of the axon whose position along the axon is given by the minimum point of the second derivative of V_m; its peak value is 2×10^{-2} $\mu\mathrm{m}^{-1}$. The integrand of equation [8.6] is the product of the blue curves with the black curves. Bottom, a summary of numerical solutions for axons with radii between 10^{-1} and 20 $\mu\mathrm{m}$. On the left, the (minimum values of) Φ_e over the value of Φ_e at a radius of 10^{-1} $\mu\mathrm{m}$ as a function of the ratio of the radii. The y-values of the red curve are the x-values to power 1.8, and those of the blue curve are to power 2.2. The various black curves show, from bottom to top, electrodes placed at 25, 50, 75 and 100 $\mu\mathrm{m}$ from the center of the axon. On the right, diagrams showing the evolution of (the negative of the minimum value of) Φ_e for a given axon radius as a function of the distance d between the electrode and the center of the axon. The radii of the axons are from bottom to top: 10^{-1}, 5×10^{-1}, 1, 10 and 20 $\mu\mathrm{m}$. For a color version of this figure, see www.iste.co.uk/clerc/interfaces1.zip*

8.2.3. *"From electrodes to tetrodes"*

One feature that is clearly visible in Figure 8.2 (bottom right) is the $1/r^2$ decay of the potential with the distance. Thus, when two neurons "of same type" are equidistant from an electrode, they will generate similar signals at that electrode. Now, if a second electrode is placed nearby but at a distinct location from the first, and if the first neuron is located "between the two", whereas the second neuron is closer to one electrode but further from the other; the first neuron will generate signals of similar amplitude at both electrodes, and the second neuron will generate a "large" signal at the first electrode and a "small" signal at the second electrode. This reasoning may be extended to greater numbers of electrodes, and explains why tetrodes are used[12]. Figure 8.3 shows how tetrodes can help to classify spikes with similar shapes and amplitudes at one recording site but distinct features at the other sites.

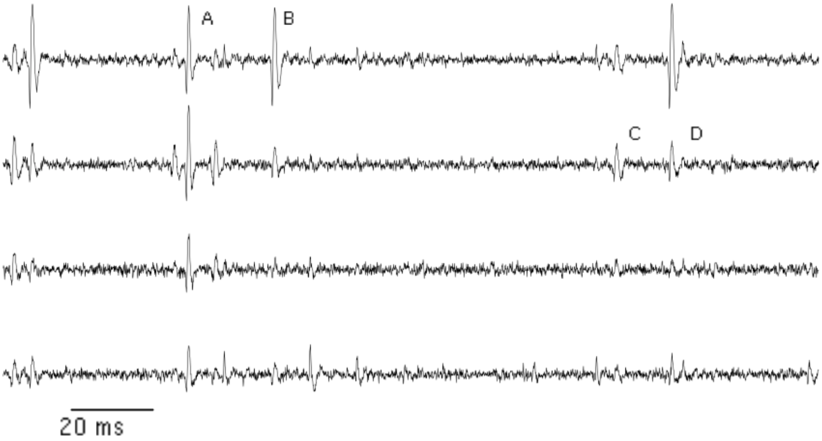

Figure 8.3. *Two hundred millisecond of data from the recording shown in Figure 8.1. A and B mark two action potentials with amplitudes and shapes that are similar at site 1, but very different at the three other sites. Similarly, C and D mark two action potentials that are similar at site 2, but different at the other sites*

[12] However, tetrodes are not always useful; for example, they serve no purpose in the antennal lobe of the cockroach *Periplaneta americana*, while in the present example, in the antennal lobe of the locust *Schistocerca americana*, they are essentially indispensable.

8.3. Spike sorting: a chronological presentation

We continue with a “chronological” presentation of the principal methods of spike sorting. This approach is not particularly synthetic, but in our opinion it introduces the various relevant problems and solutions into a concrete setting with minimal formalism. The figures in each section are drawn from actual data. A few simplifications were made, such as the use of a single recording site when four sites were available. The only differences between these examples and what is done “in practice” are technical in nature, the *ideas are the same*, and it is the ideas that are important. The data and a full step-by-step description of its analysis with the software packages R[13] and Python[14] are available on the author’s website[15].

8.3.1. *Naked eye sorting*

When the “all-or-nothing” property of action potentials in the axon was first established [ADR 22], the only tools available to neurophysiologists were recordings on paper (the oscilloscope had not yet been invented), and they had to laboriously carry out sorting with the naked eye based on amplitude (Figure 4 in [HAR 32]), in a somewhat similar fashion to the way that we analyzed the first site of Figure 8.1.

8.3.2. *Window discriminator (1963)*

Once magnetic tape recording systems had become commonplace in physiology labs, the quantity of data to be processed increased significantly, with the immediate result of inspiring certain researchers to automate the processes that they had previously been performing by hand [POG 63]. The first innovation was to construct dedicated electronic circuits. Samples were classified by the peak amplitude of their events[16] as illustrated in Figure 8.4.

[13] http://www.r-project.org/.

[14] https://www.python.org/.

[15] http://xtof.perso.math.cnrs.fr/sorting.html, at the bottom of the page.

[16] This method is still used today in some labs, especially those that need to perform sorting in real time. It is also used systematically for the audio outputs of amplifiers. Experimental researchers listen to the output of one of the electrodes when inserting them into the tissue; and the electronic circuit between the amplifier output and the speaker removes all amplitudes

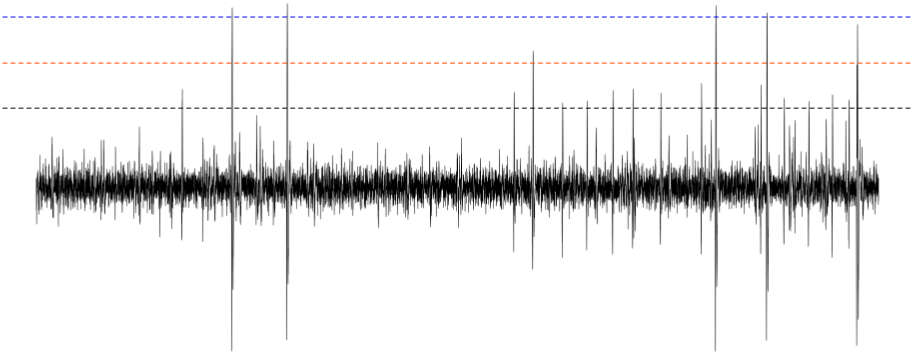

Figure 8.4. *Principle of the window discriminator. Three zones corresponding to three classes of action potentials and thus three "neurons" are defined here: small action potentials are those whose peak value is located between the black and orange dotted lines; medium action potentials have a peak value between the orange and blue lines, and large action potentials have a peak value higher than the blue line. The data were taken from the first site in Figure 8.1. For a color version of this figure, see www.iste.co.uk/clerc/interfaces1.zip*

8.3.3. *Template matching (1964)*

Physiologists soon realized that action potentials originating from two different neurons could have the same peak amplitude but different shapes (Figure 8.5(a)). This led to the introduction of a two-step method [GER 64]:

1) Two events with the same class of shape or template were identified with the naked eye, and for each pattern, a dozen or so events were averaged. These averages would subsequently serve as template estimators.

2) Each event was compared to each template by subtracting the template from the event and calculating the sum of the squares of the components of the difference vector, i.e. the *residual* vector (Figure 8.5(b)). The event is then assigned to the closest template, that is to say the template with the smallest residual vector.

below a given threshold and saturates all amplitudes above a second threshold. Since large spikes will be above the second threshold for longer than short spikes, the amplitude of each event is encoded into the duration of the sounds.

In statistics, an *estimator*[17] is a function of the data that provides an estimate of a parameter. Here, the parameters are the templates, or more concretely an ordered sequence or *vector* of 45 amplitudes (in the original article [GER 64], these vectors were defined by 32 amplitudes). Estimators are functions of the data, so the value of an estimator changes as the data changes; formally, they are *random variables*.

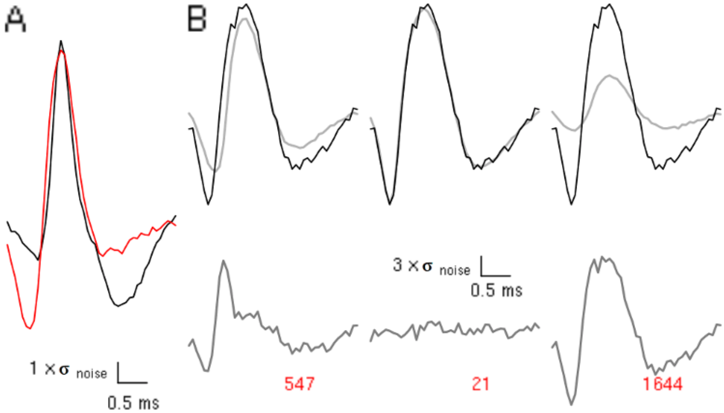

Figure 8.5. *Principle of template matching A), the templates of neurons 6 (in red) and 7 (in black) (at the fourth site). B) top, the same event (in black) and three of the 10 templates (in gray) at the first site; below, the templates were subtracted from the event, and the corresponding sum of the squares of the residues is shown in red. The event clearly matches the second template. For a color version of this figure, see www.iste.co.uk/clerc/interfaces1.zip*

8.3.4. *Dimension reduction and clustering (1965)*

Faced with the problem of low computer memory availability in physiology labs, Simon [SIM 65] had the idea that we should avoid working with the full sequence of amplitudes as required by the technique of template matching, instead restricting attention to the amplitudes measured at two carefully chosen points in time of the event (Figure 8.6(a)). These points in time were selected by observing the superimposed events together, so that the amplitudes of the different categories of spike would be distinguished as clearly as possible. This technique was able to reduce the number of

[17] See: https://en.wikipedia.org/wiki/Estimator.

parameters necessary for characterizing an event by 30 or more; graphically, we go from A to B in Figure 8.6. *Disjoint* domains in the plane representing the data were constructed by the user “by hand”, each domain corresponding to one neuron. Events were then classified according to the domain to which they belonged (Figure 8.6(c)). Once this classification had been performed on the reduced-dimension space, it is still possible, and perhaps even advisable, to return to the initial representation in order to review the results (Figure 8.6(d)). In today’s terminology, we would say that we *reduced the dimension*[18] by passing from A to B in Figure 8.6. This process of defining domains is an example of what is now known as *clustering*. These two very important aspects of high-dimensional data analysis – of which spike sorting is an example – are described in a manner that is both general and very pedagogical in the book by Hastie *et al*. [HAS 09]. The two spaces between which we have been moving, the 45-dimensional space (Figures 8.6(a) and 8.6(d)) and the 2-dimensional space (Figures 8.6(b) and 8.6(c)), are called *sample spaces*[19] by statisticians [BRE 09].

8.3.5. *Principal component analysis (1968)*

The fact that the user must choose two points at which to compare the amplitude as coordinates for the reduced space is inconvenient in the previous method. Physiologists therefore kept looking for alternative methods, more automatic and more efficient. Principal components analysis [GLA 68] was the first such alternative to be proposed, and today remains the most widely employed technique. Principal components analysis finds the subspace of desired dimension that reproduces the largest possible fraction of the variance of the sample – here, the term sample is used in the statistical sense: a set of observations/individuals randomly selected from a population. We will not discuss this method further here [GLA 76, HAS 09], but we will mention the fact that typical applications involve the intermediate step of singular value decomposition[20] of the covariance matrix of data, which we shall briefly explain in the following note.

[18] See the section on dimension reduction at: https://en.wikipedia.org/wiki/Dimensionality_reduction.

[19] The sample space is the set of all potentially observable events in an experiment. The first step of probabilistic modeling is to define this space.

[20] See: https://en.wikipedia.org/wiki/Singular_value_decomposition.

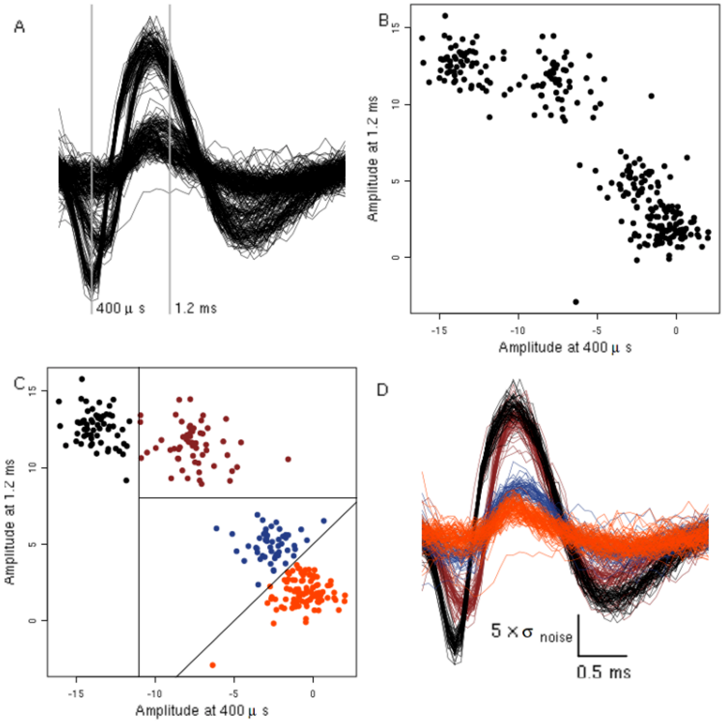

Figure 8.6. *Principle of dimension reduction and clustering. A) 267 spikes recorded at the first site. The vertical gray lines at 400 μs and 1.2 ms indicate the two chosen points in time. B) Amplitude at 1.2 ms as a function of the amplitude at 400 μs. C) Same as (B), except the boundaries of a partition have been defined manually; the spikes are colored according to the class that contains them. D) Same as (A), except the spikes are colored according to the class that contains their projections. For a color version of this figure, see www.iste.co.uk/clerc/interfaces1.zip*

NOTE.– The covariance matrix is constructed from the matrix D of data, whose rows are the events. In Figure 8.6, D is a matrix with 267 rows – there are 267 events – and 45 columns – each event e_i is specified by an ordered set of 45 amplitudes $(e_{i,1}, \ldots, e_{i,45})_{i=1,\ldots,267}$. Each of the action potentials in Figure 8.6(a) corresponds to one row of the matrix D. The covariance matrix is obtained by subtracting the average row $(\overline{e}_j)_{j=1,\ldots,45}$ where $\overline{e}_j = \sum_{i=1}^{267} e_{i,j}/267$ from each of the rows $(e_{i,1}, \ldots, e_{i,45})_{i=1,\ldots,267}$, which yields the matrix M whose entries are given by $M_{i,j} = e_{i,j} - \overline{e}_j$. The entries of the covariance matrix V are then given by $V_{i,j} = \sum_{k=1}^{267} M_{i,k}\, M_{k,j}/267$, or, written as matrix multiplication: $V = M^T M/267$, where M^T is the transpose of M (See https://en.wikipedia.org/wiki/Covariance). The result of principal component analysis on the set of data in the previous section is shown in Figure 8.7. Figure 8.7(a) shows both the average event (in black) and the average event plus the first (in red) and second (in blue) principal component multiplied by 10. We see that events whose projections onto the first component have high values differ from the average in amplitude but not in shape, and that events whose projections onto the second component have high values differ in shape but not in amplitude. The reader should note (Figure 8.7(b)) that the multiplicative factor of 10 is the same order of magnitude as the observed values. Figure 8.7(b) corresponds to Figures 8.6(b) and 8.6(c) and shows that it is easy to define domains by reducing the dimension along the principal components. Nevertheless, we should note that performing principal component analysis requires a certain amount of (computer) memory, the absence of which was precisely what originally motivated Simon [SIM 65] to introduce the idea of dimension reduction. These memory constraints have now long since disappeared.

8.3.6. *Resolving superposition (1972)*

Since the deviations of the extracellular potentials of action potentials are of the order of the millisecond, we can expect to observe instances of superposition[21] similar to those shown in Figure 8.8(a) whenever sufficient neurons are registered by the recording[22]. This phenomenon was characterized in the early 1970s, and solutions based on "manual" template matching were suggested [PRO 72]. Clearly, as shown in Figure 8.8, resolving superposition requires the templates to have been estimated: superposition cannot be resolved simply by considering the projection of the data onto a subspace like in the previous two sections. Today, the most

[21] In the literature, the terms "collision" and "interference" are also used to describe this phenomenon.

[22] If ν is the average discharge frequency of K neurons in a recording, and Δ is the typical duration of an action potential and if we assume that neuron discharges may be modeled sufficiently accurately by a Poisson distribution, then the probability of there being zero action potentials within a window of duration Δ is $\exp -K\nu\Delta$, the probability of there being exactly one is $K\nu\Delta \exp -K\nu\Delta$ and the probability of there being at least two is $1 - (1 + K\nu\Delta) \exp -K\nu\Delta$; the frequency of superposition among windows containing at least one event is the ratio of these last two values.

commonly used methods of spike sorting are derived from methods that combine dimension reduction and clustering; since these techniques are not capable of resolving superposition, it appears that the majority of published results that rely on sorting simply ignore superposition.

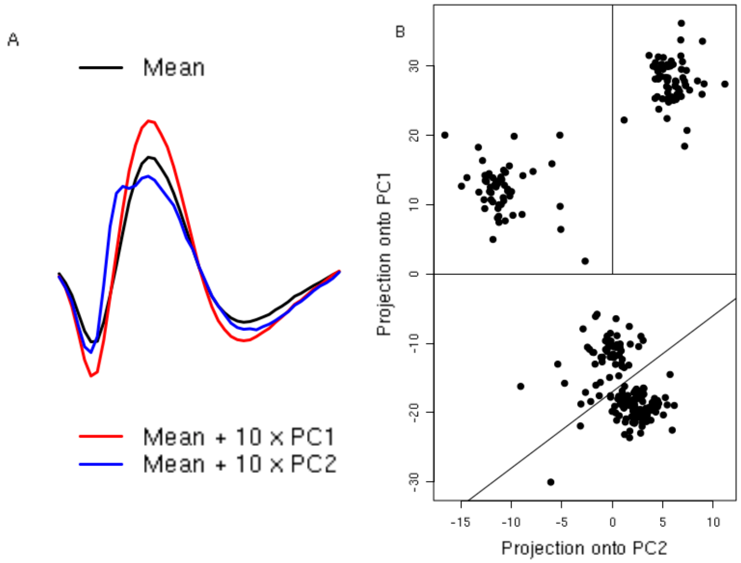

Figure 8.7. *Principal component analysis. A) In black, the average of the 267 action potentials recorded at the first site; in red, the same average plus 10 times the first principal component; in blue, the same average plus 10 times the second principal component. B) Two hundred sixty-seven events projected onto the plane defined by the first two principal components. Domains that yield the same classification as in Figure 8.6 have been added by hand. For a color version of this figure, see www.iste.co.uk/clerc/interfaces1.zip*

8.3.7. *Dynamic amplitude profiles of action potentials (1973)*

Until the early 1970s, recordings of isolated axons or nerves were very common, especially in invertebrates. In this type of recording, the "all-or-nothing" property of action potentials effectively holds, even during high-frequency firing. But as cortical recordings in vertebrates became more commonplace, a new problem particular to these subjects soon presented itself: the dynamic character of the amplitude profiles (and sometimes the shape) of the action potentials emitted by a neuron during high-frequency or *burst* discharges, as shown by Figure 8.9(a). The solution suggested by Calvin [CAL 73] requires manually processing the data, and relies on a “relatively” stable combination of amplitude reduction and interspike intervals during bursts (Figure 8.9(c)). Note how these amplitude dynamics introduce additional obstacles for spike sorting (Figure 8.9(b)).

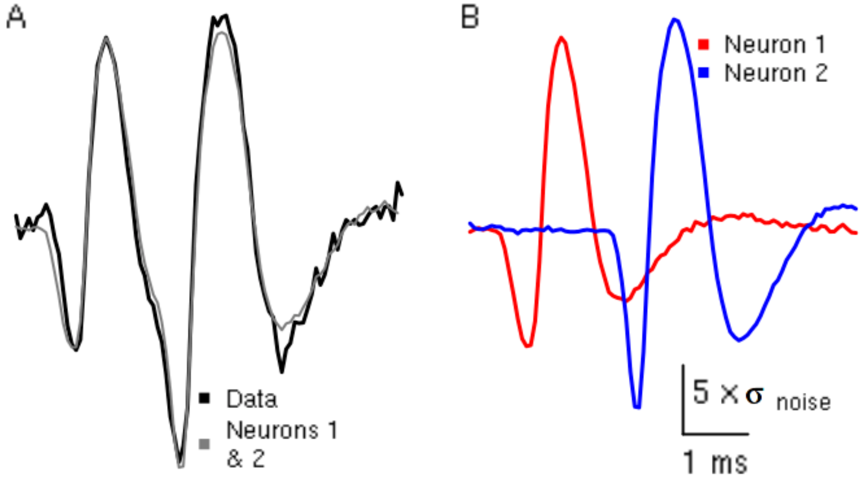

Figure 8.8. *Resolving superposition. A) In bold black, an event that does not correspond to any of the templates/neurons; in gray (thinner) the sum of the two patterns – associated with neurons 1 and 2 of our classification – shown in B). The scales of the two graphs are identical. For a color version of this figure, see www.iste.co.uk/clerc/interfaces1.zip*

8.3.8. *Optimal filters (1975)*

Attempts to perform multiple recordings along the nerve of a marine invertebrate led Roberts and Hartline [ROB 75] to suggest a method capable of automatically decomposing instances of superposition. Their method may be viewed as an extension of the template matching method; the idea is to construct one filter per neuron such that the filter is maximal when an action potential emitted from the neuron *for which the filter was constructed* is present in the data, and minimal or zero when noise or emissions from another neuron are running through the signal. The filters are linear, so that if action potentials of two or more neurons are present with a small offset in time, such as in Figure 8.8(a), the output of the two filters should display spikes with the same offset in time. Similarly to the technique of template matching, the method assumes that we have previously estimated the characteristic shapes/patterns associated with each neuron, at each recording site if multiple sites are in use. The construction of optimal filters is slightly too complicated to be fully explained here ([ROB 79] has all of the details), but we will illustrate the idea with the example of *matched filters*[23]. The characteristic shapes of the neurons obtained at each of the sites (three templates corresponding to three different neurons are shown in gray in the upper section of Figure 8.5(b)) are represented by a set of vectors, with one vector per recording site. Each vector has the same number of elements, corresponding to the number of sample points in the template – this method works best with lengthy templates that start at zero and return to zero, whereas in general the method of template matching works well even with shorter templates – in the case shown in Figure 8.10, for the second neuron, we have 130 points per template at sites 1 and 4:

$$\mathbf{m}_2 = \begin{pmatrix} m_{2,1} \\ m_{2,4} \end{pmatrix} = \begin{pmatrix} m_{2,1,1}, \dots, m_{2,1,130} \\ m_{2,4,1}, \dots, m_{2,4,130} \end{pmatrix}$$

To construct a *matched filter* from these two vectors, we begin by subtracting from each $m_{2,i,j}$ the average at the corresponding site: $m_{2,i,\bullet} = \sum_{j=1}^{130} m_{2,i,j}/130$ to obtain $f_{2,i,j} = m_{2,i,j} - m_{2,i,\bullet}$, and then we normalize so that the scalar product of the filter $\mathbf{f}_2$ with the original template $(\sum_{i\in\{1,4\}} \sum_{j=1}^{130} m_{2,i,j} f_{2,i,j})$ is equal to one. In the suboptimal case of a *matched filter*, the filters are therefore just normalized versions of the templates. If we write the data to which the filter will be applied in the following form:

$$\begin{pmatrix} \dots, d_{1,k-2}, d_{1,k-1}, d_{1,k}, d_{1,k+1}, d_{1,k+2}, \dots \\ \dots, d_{4,k-2}, d_{4,k-1}, d_{4,k}, d_{4,k+1}, d_{4,k+2}, \dots \end{pmatrix}$$

then the output $F_{2,k}$ of filter 2 at "time" k is given by the expression:

$$F_{2,k} = \sum_{i\in\{1,4\}} \sum_{j=1}^{130} f_{2,i,j} d_{i,k+j-J} \,,$$

[23] See: http://en.wikipedia.org/wiki/Matched_filter.

where J is the position of the peak within the template $\mathbf{m}_2$. The implementation of this method, and the way that it can automatically resolve superposition, are demonstrated in Figure 8.10. *Matched filters* are suboptimal because their "interference", that is to say the output of the filter for templates that it was not designed to match, has not been optimized. The secondary peak in the output of filter 2 shown in Figure 8.10(B) is one such example of interference. The method of filter construction presented in [ROB 79] reduces this problem significantly; nevertheless, if too many of the characteristic shapes are too similar it will not be possible to fully eliminate the interference.

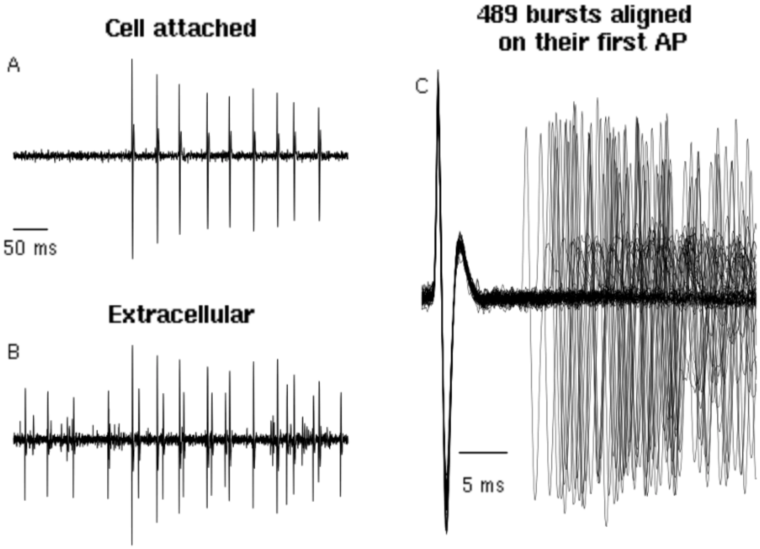

Figure 8.9. *Dynamic amplitude profiles of action potentials A), example of a burst, somatic recording in a "cell-attached" arrangement of a Purkinje cell in a slice of cerebral cortex of a rat. Note how the amplitude of the action potentials is diminished during the burst. B) Simultaneous extracellular recording (recordings taken by Matthieu Delescluse). Three neurons, including the one recorded in cell-attached mode, register on this recording. Notice how the action potentials of a tonically active neuron have a similar amplitude to the action potentials of the reference neuron (also recorded in a cell-attached arrangement) at the end of the burst. C) Four hundred eighty-nine recorded bursts over the course of 1 min aligned by their first action potential. The details of the recordings and data processing steps are given in [DEL 06]*

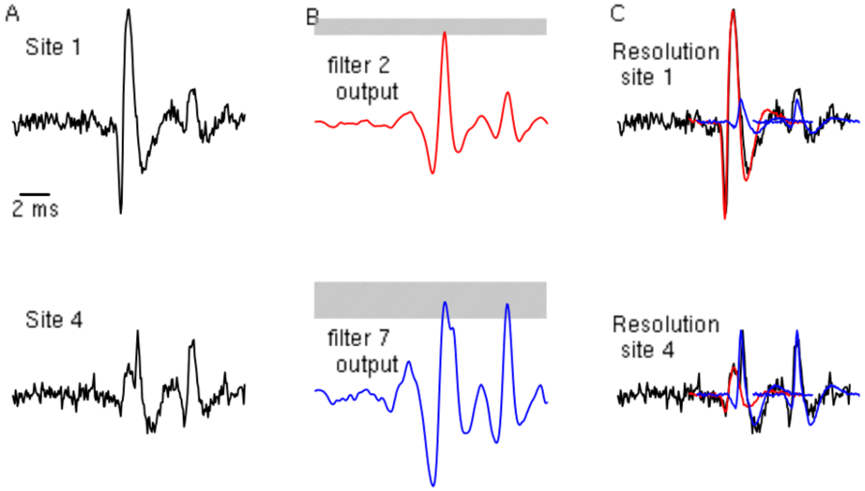

Figure 8.10. *Matched filters A), a subset of the data from the grasshopper example, at two of the four sites of recording. B) Corresponding output of the matched filters constructed from the templates of neurons 2 and 7. The gray bands are 99% confidence bands obtained by superimposing "noise events" – segments of raw data between the two detected action potentials – and the template of each of the two neurons before applying the filter. C) The resolved signals obtained from the filter outputs. For a color version of this figure, see www.iste.co.uk/clerc/interfaces1.zip*

8.3.9. *Stereotrodes and amplitude ratios (1983)*

The most direct and possibly still the most effective method of sorting spikes with dynamic amplitude profiles was suggested by McNaughton *et al.* [MCN 83]. It is perhaps not so much a method of analysis, but rather a recording technique: stereotrodes (two recording sites in close proximity, as suggested in the original article) or tetrodes (four sites in close proximity [GRA 95]). The motivation for this method is presented in perfect clarity in the penultimate paragraph of their introduction:

> "The method described in the present report is based on the fact that the size of the extracellular action potential varies inversely with the distance of the recording electrode from the current generator. In theory, a closely spaced tetrahedral array of recording electrodes with tips sufficiently close together to record

> signals from overlapping populations of neurons should permit the unique identification of all neuronal spikes that exceed the noise level. This is so since each cell would generate a unique point in the three-dimensional data space whose axes are defined by the spike height ratios of channels 1 and 2, 2 and 3, and 3 and 4. Note, that since the discrimination is based on amplitude ratios, the problem of intrinsic variation in spike amplitude such as occurs during the complex spike burst of hippocampal pyramidal cells is, in principle, solved".

The data recorded in slices of the cerebral cortex of a rat, which we previously used in Figure 8.9, will once again serve to illustrate the principle of amplitude ratios. Figure 8.11 shows 200 ms of data recorded at the two sites (separated by 50 μm) of a stereotrode. The action potentials from the "reference" cell in this last figure are marked with vertical gray dotted lines. Action potentials from a different cell that fires "in pairs" (and sometimes in triplets) with strongly characteristic amplitude dynamics are also marked with vertical gray lines.

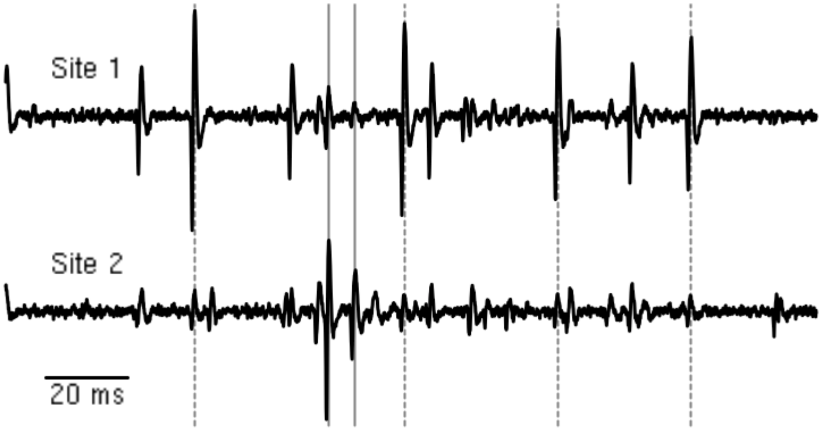

Figure 8.11. *Data from a stereotrode 200 ms of extracellular data recorded along the surface of the cell bodies of Purkinje cells – cerebral cortex slice of a young rat, same dataset as in Figures 8.9. The vertical gray lines mark two action potentials of a cell firing in pairs (with a strongly dynamic amplitude profile). The vertical gray dotted lines show the action potentials of the burst-firing cell if Figure 8.9*

After detecting the action potentials by identifying the local maxima above a certain threshold, the peak amplitudes of each spike are obtained, and each action potential is represented in Figure 8.12 (left) as a point on a plane (sample space) whose axes are given by the peak amplitude at the second site (horizontal) and the peak amplitude at the first site (vertical). Each point is assigned an angle by calculating the arctangent of the amplitude ratio. Calculating the amplitude ratio is always a somewhat sensitive operation, because dividing two noisy values increases the error. In order to avoid excessively large errors, we performed regression on the amplitudes near the peak at site 1 (5 points on each side of the peak) as a function of the corresponding amplitudes at site 2, neglecting the constant term. The estimated density of the angles is shown in Figure 8.12 (right). Thus, we obtain well-defined peaks, which may be used to define angular domains corresponding to domains of amplitude ratios.

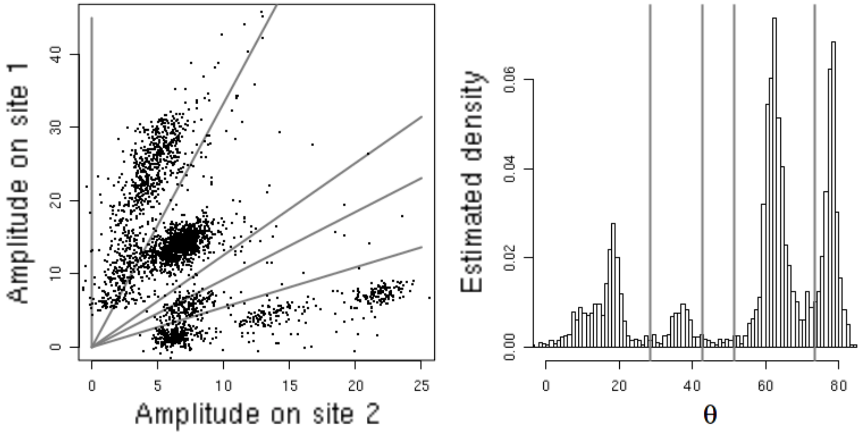

Figure 8.12. *Amplitude ratios. On the left, the (peak) amplitude at site 1 as a function of the (peak) amplitude at site 2 for the action potentials detected in the dataset in Figure 8.11. The units of the axes are standard deviations of the noise. The gray lines correspond to the angular domains defined in the right-hand section. On the right, the distribution of the θ angles estimated using the tangent obtained by regressing the 10 amplitude values in the neighborhood of the peak at site one as a function of the 10 amplitude values in the neighborhood of the peak at site 2 (neglecting the constant term). The vertical gray lines were placed “with the naked eye” to partition the angles into different categories*

At this point, we may choose between two strategies: we could perform clustering by initially ignoring the amplitude ratios and later merging classes located within the same angular domain if, after merging, a refractory period is indeed visible in the distribution of the intervals between action potentials; alternatively, we can perform clustering separately on the angular domains, merging classes so long as there remains a visible refractory period. For the data of the given example, if we restrict attention to the largest events, a classification based solely on the angular domains will be sufficient, as shown in Figure 8.13.

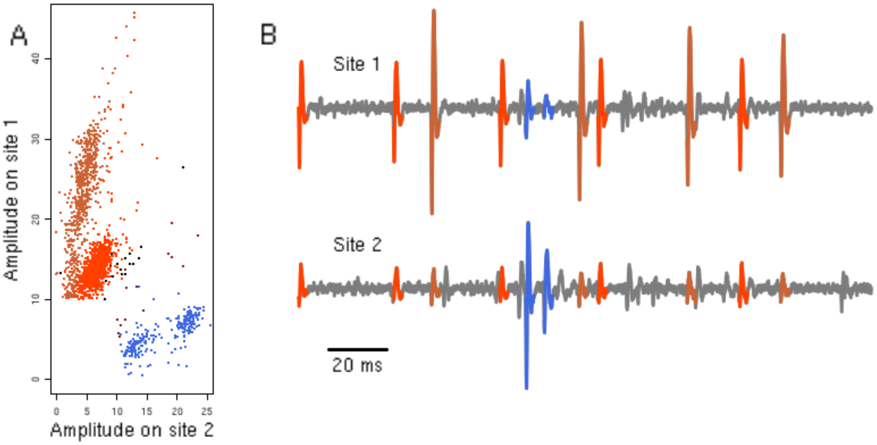

Figure 8.13. *Classification based on amplitude ratios. A) Left section of Figure 8.12, with points colored by amplitude ratio. Note that only spikes with peak amplitude larger than 10 at one or more sites have been retained. B) Identical to Figure 8.11 with spikes colored by amplitude ratio. The cell firing in pairs has been correctly identified (in blue), and so has the cell firing in bursts (in brown) and the cell firing "constantly" (in orange). For a color version of this figure, see www.iste.co.uk/clerc/interfaces1.zip*

8.3.10. *Sampling jitter (1984)*

One specific difficulty arises with the technique of data sampling. The data are physically saved in the form of sequences (or vectors) of amplitudes – values of the amplitude at uniformly separated points in time – whereas the original true data were *continuous*. In an ideal situation, without any recording noise, we would expect the position of two consecutive action potentials generated by the same neuron to be shifted relative to the sampling times (technically, we would usually use the term *phase* rather than position for this), as shown in Figure 8.14(a). If analysis is performed directly on the sampled data, for example an attempt to resolve superposition by subtracting the closest-fitting template (Figure 8.8), new events may be unintentionally introduced as a result, as shown in Figure 8.14(b1). In this example, the event sampled at the bottom of Figure 8.14(a) was used as a template and subtracted from the event sampled at the top of Figure 8.14(a). The peak amplitude of the difference is equal to five times the standard deviation of the noise, which means that it would be identified as a new spike, as our detection threshold was chosen to be four times the standard deviation of the noise for these data. Another way of visualizing the consequences of sampling jitter is by simulating, using a continuous template – more precisely a template defined by a continuous *function* – noisy sampled data with and without jitter. The jitter is simulated by a uniformly distributed random variable taking values in an interval of $-1/2$ to $+1/2$ of the sampling period. The template is then subtracted from the simulated data, and the sum of the squares of the residues is calculated (as we did earlier for Figure 8.5(b)). The distribution of the sum of the squares of the residues is shown in Figure 8.14(b2). We see that jitter can have an effect on the variability that is of the same order of the effect of noise. This effect depends on the sampling rate and on the shape of the template, as explained in [POU 14]. The usual strategy for counteracting this problem is to sample at high frequencies, but this is only feasible when relatively few channels are recorded simultaneously. Another option is to numerically resample by application of the Nyquist–Shannon theorem [POU 02]. It is also possible to effectively correct for jitter using the method suggested in 1984 by McGill and Dorfman [MCG 84] – using Fourier transforms – or using a Taylor–McLaurin series expansion [POU 14]. Finally, for purposes of resolving superposition – in the author's experience – it seems to be the case that the effect of jitter is less noticeable when using filters (section 8.3.8) as compared to subtraction-based methods (section 8.3.6), although these apparent differences are yet to be documented in a "serious" study.

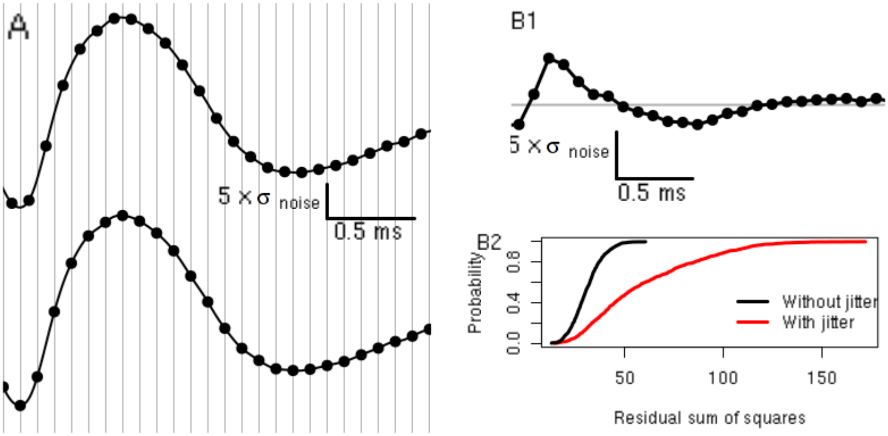

Figure 8.14. *Sampling jitter. A) Two action potentials from the same neuron sampled with two different phases; the points correspond to the digitally sampled amplitudes, and the continuous line corresponds to the data pre-sampling. B1) Illustration of the jitter effect –* without recording noise *– with template matching; here, the template is the sampled version of the figure at the bottom of A, and the event is the sampled version of the figure shown above. The "bottom" amplitudes are subtracted from the top amplitudes (which precede them by half of an amplitude period). B2) A simulation comparing 1,000 events with and without uniform jitter on +/- half of a sampling period with white recording noise following a normal distribution (here, the events were defined by sequences of 30 amplitudes). For a color version of this figure, see www.iste.co.uk/clerc/interfaces1.zip*

8.3.11. *Graphical tools*

Since the late 1980s, there have been spectacular improvements in the computational power of computers, with the introduction of *interactive* methods of visualization, the first and foremost of which is most certainly the program XCLUST developed by Wilson[24]. These methods involve the systematic application of the techniques of dimension reduction as discussed in section 8.3.4; they allow multiple projections to be visualized simultaneously. Thus, instead of working with only the first two principal components (Figure 8.7), we are able to work with four or more, and compare the graphs of the projections onto planes defined by pairs of any two of the

[24] The latest versions of this program are available on github: github.com/wilsonlab/mwsoft64.

principal components (first and second, first and third, etc.). Figure 8.15 shows a screenshot of the software package `GGobi` – free of cost and open source[25] – showing an example of one such matrix of projections[26]. This figure attempts to show the interactivity of the program as much as possible. The "active" panel or graph is at the top-left of the figure (with a black border). The little light-blue square is the "paintbrush", which the user is free to move using the mouse. Each point, initially magenta colored, becomes blue on the active graph once it is selected by the blue square as well as the corresponding points on all of the other graphs in the matrix. The technique of coloring in parallel equivalent points on multiple graphs is called *brushing*; see [CLE 93, p. 294] and [COO 07]. Today, this is how most spike sorting is performed. The program that we are showcasing here, `GGobi`, is capable of providing even more sophisticated (and extremely useful) dynamic visualizations, such as the "grand tours" introduced by Asimov [ASI 85]. In our experience, although `GGobi` is not sufficient[27] for spike sorting, it is, nevertheless, the most important software package for this task.

8.3.12. *Automatic clustering*

With the development of interactive graphical methods since the late 1980s, the development, or the adoption, of automatic or semiautomatic clustering methods has been the greatest focus of spike sorting "methodologists". Indeed, the problem with the methods presented up to this point is that they require "significant" effort from the researcher performing data analysis. "Template matching" (section 8.3.3) and "filtering" (section 8.3.8) require the templates and filters to be estimated, and the methods combining dimension reduction and clustering (sections 8.3.4, 8.3.5 and 8.3.11) require the classes or event groups to be defined directly by the user. These "heavy" tasks have the following two effects on the analysis:

[25] Available for download free of cost for `Linux`, `Windows` and `Mac` at http://www.ggobi.org/.

[26] This figure was prepared using the locust dataset, but unlike Figures 8.6 and 8.7, all of the events and all four recording sites were included – for clarity, the previous figures were prepared with subsets of the events from one single recording site.

[27] Because resolving superposition is not possible, at least not easily, after performing dimension reduction.

1) the analysis becomes time intensive;

2) the analysis becomes difficult to reproduce[28].

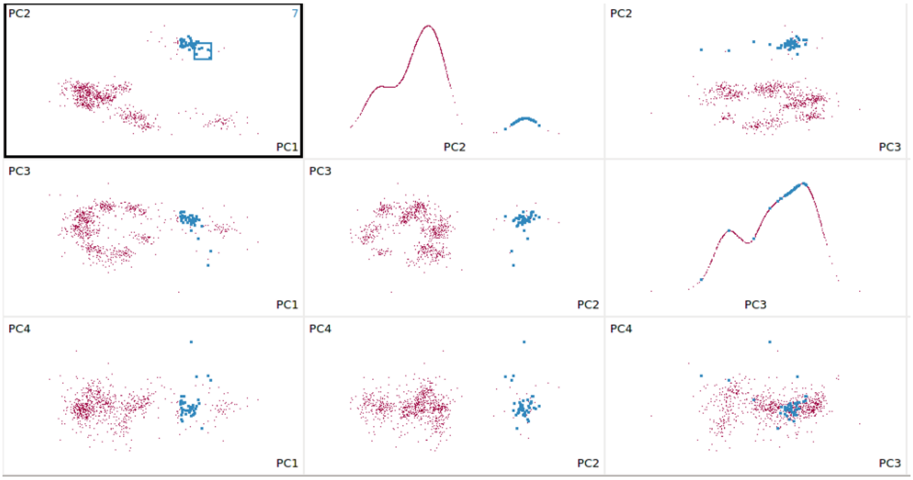

Figure 8.15. *Matrix of scatter plots showing the projections of a set of events onto the planes defined by pairs taken from the set of the first four principal components: a "screenshot" of the software package* `GGobi`*. The "diagonal" graphs (first row, second column and second row, third column) are the smooth estimates of the densities of the projections of the events onto the second (first row, second column) and third (second row, third column) principal components. For a color version of this figure, see www.iste.co.uk/clerc/interfaces1.zip*

There is therefore a strong demand for automatic methods, which has in the past inspired a large number of publications (and still does to this day). At the risk of angering a fair few of our colleagues, we wish to venture the opinion that most of the obstacles encountered in the context of (the clustering stage of) spike sorting are addressed in sufficient depth by the two most common statistical methods for this type of problem:

[28] Reproducibility fails at two different levels: two different people analyzing the same dataset will usually not define the same classification as illustrated in [HAR 00]; and one *same* person analyzing the *same* dataset 6 months later will usually not define the same classification twice.

1) the *k-means* algorithm[29];

2) *Gaussian mixture models* (or GMM)[30] modified by the *expectation–maximization* or *[EM] algorithm*)[31].

The k-means algorithm is easy to specify (and implement):

– *Choice of number of components*: the number of classes k to include in the model[32] is chosen by the user;

– *Initialization*: k events, which we shall call *centroids*, are chosen at random among the n observed events;

– *Distance calculation*: the (Euclidean) distance of each event from each of the k centroids is calculated;

– *Event assignment*: each event is assigned to the centroid to which it is closest;

– *Centroid update*: the *updated* position of each centroid is calculated as the average of the events that "belong" to that centroid;

– *Iteration*: return to the *distance calculation* step until a maximal number of iterations – chosen beforehand – is reached, or another stopping condition[33] is satisfied.

– *Results*: the final values of the centroids are the "templates", the final assignments yield the classification and the "total variance"[34] is calculated.

This procedure is repeated multiple times (10–50 times) with *different initializations*; the final result is taken to be the instance with the smallest final total variance. The way that the algorithm works is illustrated in Figure 8.16. The data are from Figure 8.15, but to make the illustration easier to read, one

[29] See: https://en.wikipedia.org/wiki/K-means_clustering.

[30] See: https://en.wikipedia.org/wiki/Mixture_model.

[31] See: https://en.wikipedia.org/wiki/Expectation-maximization_ algorithm.

[32] In practice, observing the data using "dynamic" modes (rotations and "grand tours") in GGobi allows k to be chosen. We will discuss automatic methods at a later point.

[33] An example of a stopping condition is when all distances between two between consecutive values for each of the centroids are below a chosen threshold.

[34] Each centroid is subtracted from each of its assigned events (vector subtraction) and the squares of the (Euclidean) norms of these differences are summed. This sum is denoted the "total variance".

single projection (onto the plane defined by the first and third principal components) was used and the events from three of the 10 neurons – which were identified when the analysis was performed "properly" – were omitted.

The EM algorithm for a GMM adds an extra layer of formalism to the k-means algorithm: a probabilistic model of data generation is therefore *explicitly* assumed. Each observation $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n$ is viewed as the realization of a random variable $\mathbf{Y} \in \mathbb{R}^p$ whose distribution is known up to a *finite* number of parameters. In the case of a *Gaussian mixture*, the density of $\mathbf{Y}$ may be written as:

$$p\left(\mathbf{Y} = \mathbf{y}; \theta_k\right) = \sum_{j=1}^{k} \pi_j \, \phi(\mathbf{y}; \mu_j, \Sigma_j)\,, \qquad [8.8]$$

where θ_k is the set of model parameters,

$$\theta_k = \{\pi_j, \mu_j, \Sigma_j\}_{j=1,\ldots,k}\,,\ 0 \leq \pi_j \leq 1\,,\ \sum_{j=1}^{k} \pi_j = 1\,, \qquad [8.9]$$

and where $\phi(\ ; \mu, \Sigma)$ is the density of a multidimensional normal (or Gaussian) distribution:

$$\phi(\mathbf{y}; \mu, \Sigma) = \frac{1}{(2\pi)^{p/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{y} - \mu)^T \Sigma^{-1} (\mathbf{y} - \mu)\right)\,, \qquad [8.10]$$

where μ is the mean, a vector in $\mathbb{R}^p$, Σ is the covariance matrix[20], $|\Sigma|$ is the determinant of Σ and the superscript T is the *transpose*. The unknowns of the mixture distribution are the weights π_j – there are only $k - 1$ independent values – the k means μ_j and the k covariance matrices Σ_j. With this setup, the EM algorithm for a GMM is only slightly more complicated than the k-means algorithm. In the general case where each neuron/aggregation has its own covariance matrix, it may be stated as follows:

– *Initialization*: k events are randomly chosen from the n observed events, which are taken as the k $\mu_j^{(0)}$. The $\pi_j^{(0)}$ are typically all initialized with identical values equal to $1/k$, and the $\Sigma_j^{(0)}$ are also initialized identically as diagonal matrices with elements equal to the variance of the noise;

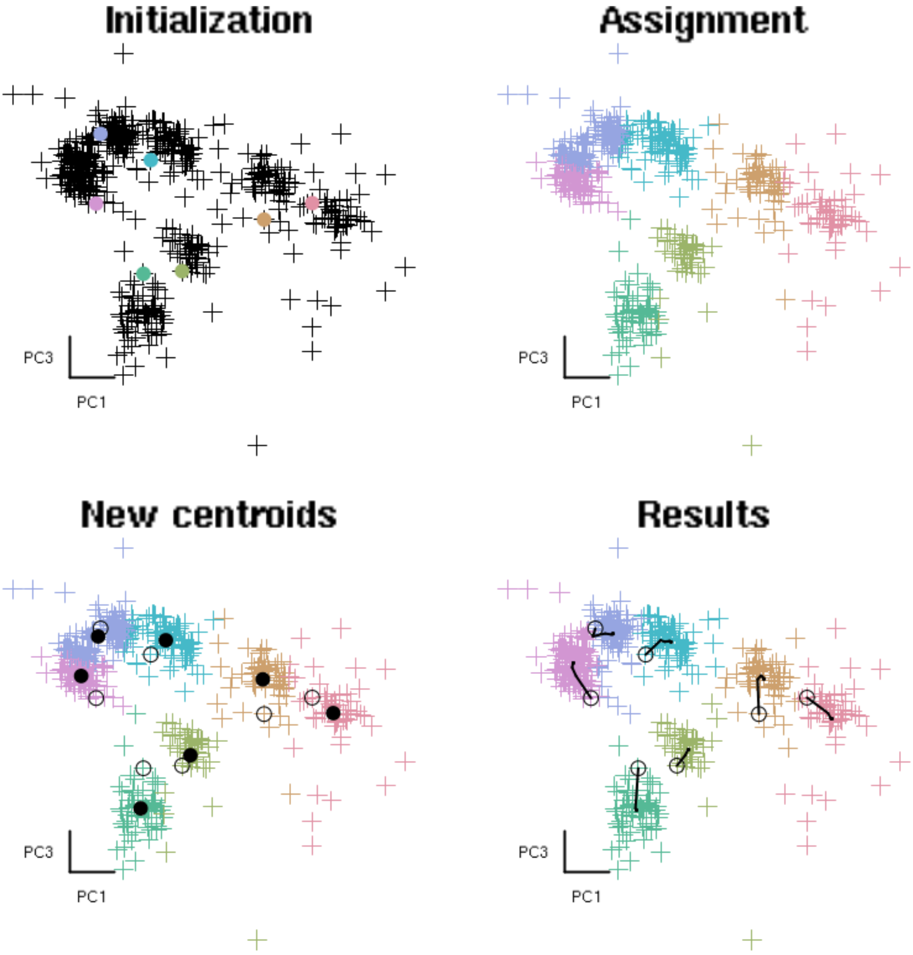

Figure 8.16. *K-means algorithm. Initialization: seven events (colored discs) were randomly chosen from the set of events (black crosses). Assignment: the distance between each centroid and each event was calculated, and each event was assigned to the nearest centroid. Note how the concentration of points centered around the sky blue section is partitioned into segments such that its edges are assigned to its neighbors, colored magenta and turquoise. New centroids: the updated positions of the centroids (black discs) are calculated; the old positions are shown as circles. Results: after 20 iterations of the algorithm, the final assignments are obtained. The trajectories of the centroids are shown in black, and the initial positions are shown as circles. For a color version of this figure, see www.iste.co.uk/clerc/interfaces1.zip*

– *Calculation of relative likelihoods*: the relative likelihood $p_{i,j}$ that the jth of the k centroids generated the ith event is calculated:

$$p_{i,j} = \pi_j^{(l)}\, \phi(\mathbf{y}_i; \mu_j^{(l)}, \Sigma_j^{(l)})$$

– *Assignment of responsibilities*: the “responsibility” $t_{i,j}$ of each of the centroids j for each of the events i is obtained by normalizing the relative likelihoods:

$$t_{i,j} = p_{i,j} / \sum_{m=1}^{k} p_{i,m}$$

– *Update of parameters*: new parameter values are obtained for each centroid by averaging each event weighted by the responsibility of the corresponding centroid:

$$\pi_j^{(l+1)} = \sum_{i=1}^{n} t_{i,j}/n\,,$$

$$\mu_j^{(l+1)} = \left(\sum_{i=1}^{n} t_{i,j}\, y_i \right) / \sum_{i=1}^{n} t_{i,j}$$

and

$$\Sigma_j^{(l+1)} = \left(\sum_{i=1}^{n} t_{i,j}\, (y_i - \mu_j^{(l+1)})(y_i - \mu_j^{(l+1)})^T \right) / \sum_{i=1}^{n} t_{i,j}$$

– *Iteration*: return to step *calculation of relative likelihoods* until a maximal number of iterations – chosen beforehand – has been performed, or another stopping condition is satisfied;

– *Results*: the results are given by the latest values for the parameters and the responsibilities.

Analogously to the *k*-means algorithm, this procedure is repeated multiple times (from 10–50 times) with *different initializations*; the final result chosen from the instance of the algorithm that produces the greatest final probability density – or likelihood, as explained in the next section – for the given dataset[35]. The μ_j give estimates for the centroids. The responsibilities may be utilized in two different ways: the first approach is to assign each event to the centroid with the greatest responsibility for that event; the second is to record the responsibilities as they are and perform all subsequent estimations (histograms of intervals between action potentials, cross-correlograms between neurons, etc.) by taking averages weighted by the responsibilities, as explained in section 5.4 of [POU 05]. Possible (and widely used) simplifications of the `GMM` as specified above include taking the π_j to be identical for all neurons – which amounts to assuming that they all fire with identical frequency – and taking the Σ_j to be identical – which amounts to assuming that each individual neuron always generates spikes of the same shape or template; in other words, there are no dynamic shape profiles (sections 8.3.7 and 8.3.9). By combining these last two constraints, we obtain a version of the *k*-means algorithm that allows for partial assignments in the assignment step. On the dataset in Figure 8.16, regardless of the version of `GMM` chosen, the algorithm produces a classification identical to the classification given by *k*-means, assuming that the same number of classes/neurons is used. In practice, `GMM` with `EM` is preferred over *k*-means when the concentrations of points visualized with `GGobi` have different shapes and, most importantly, when they partially overlap. When they overlap, estimating the position of the centroids (the μ of a `MMG`) will be more reliable, which is important when using these methods of automatic clustering as a preamble to a classification based on template matching (section 8.3.3) or filtering (section 8.3.8).

The theoretical basis for methods of automatically choosing the number of classes – the k parameter in the above – is the concept of penalized likelihood, and are discussed in Chapter 7 of [HAS 09]. The *likelihood* is simply the probability density of the observations (and the log-likelihood is its logarithm), except that the role of the observations and the parameters have been switched; for example, in the case of the `GMM` considered above:

$$l(\theta_k; \mathbf{y}_1, \ldots, \mathbf{y}_n) = \sum_{i=1}^{n} \log \left\{ \sum_{j=1}^{k} \pi_j \, \phi(\mathbf{y}_i; \mu_j, \Sigma_j) \right\} . \qquad [8.11]$$

[35] We would usually calculate the final log-likelihood instead: $\sum_{i=1}^{n} \log \left(\sum_{j=1}^{k} \pi_j \, \phi(\mathbf{y}_i; \mu_j, \Sigma_j) \right)$.

The log-likelihood is *a function of the parameters, assuming that the data are fixed*. A major mathematical result in this area of statistics is that by choosing the estimator $\widehat{\theta}$ of θ to be the argument that maximizes equation [8.11] *for a fixed number of classes*, we achieve an "optimal" value[36]. The fact that the sequence $\theta^{(l)}$ generated by the EM algorithm converges to $\widehat{\theta}$ (assuming some fairly general conditions) is another important mathematical result. Now, in the case of a GMM, we can immediately see that as k increases and the diagonal elements of the Σ_j decrease, the log-likelihood becomes infinite, for example if we take the number of classes to be equal to the number of observations and set $\pi_j = 1/n$ and the μ_j equal to the observations (one centroid per observation). In other words, if we attempt to maximize the likelihood while allowing the number of classes to vary, without setting a lower bound (> 0) for the diagonal elements of the Σ_j, then the likelihood is maximized by a model with as many classes as there are observations, where each centroid is equal to one of the observations and where the covariance matrices are degenerate with zeroes along the diagonal. If the data are being continually recorded, it is in principle possible to estimate the covariance matrix of the noise (similarly to [POU 02]) and to use this estimation as a constraint for the covariance matrices in each of the classes: the elements of the covariance matrices must be greater than or equal to the corresponding elements in the covariance matrix of the noise. Interestingly, this approach does not seem to have ever been pursued. Instead, more general statistical methods are typically used; these methods do not assume that it is possible to independently estimate the noise level, penalizing the likelihood by a term proportional to the "complexity" of the model – in other words, the number of parameters. This approach leads us to minimize the *Akaike information criterion* (AIC)[37]:

$$\mathrm{AIC}(k) = -2\,l(\widehat{\theta}_k) + 2\,d\,, \tag{8.12}$$

[36] Optimal in the sense that if the data were indeed generated by a mixture of Gaussian models, and if the number of observations n tends to infinity, then the random variable $\widehat{\theta}$ will converge to θ and has the smallest possible variance.

[37] See: https://en.wikipedia.org/wiki/Akaike_information_criterion.

where $\widehat{\theta}_k \in \mathbb{R}^d$ maximizes equation [8.11] and d is the dimension of the parameter space, which is a function of k. Another even more commonly used criterion is the *Bayesian information criterion* (BIC)[38]:

$$\mathrm{BIC}(k) = -2\, l(\widehat{\theta}_k) + d\, \log n\,. \qquad [8.13]$$

In the light of the discussion above, as k increases, so too does $l(\widehat{\theta}_k)$, and the first terms of the AIC and the BIC decrease; it is clear that the terms $2d$ and $d \log n$ will counteract this decrease, as they themselves increase with k. Thus, the BIC penalizes complex models more strongly than the AIC. In practice, both criteria overestimate the number of classes/neurons. This is largely due to the fact that events can overlap when clustering is performed (sections 8.3.6 and 8.3.8); these instances of superposition are not correctly accounted for in mixed models (see the remark about this at the end of section 8.3.11). It would clearly be desirable to perform a comparison of these models based on complete datasets specifically including information about instances of superposition, not just at the clustering stage, but this has not yet been pursued to our knowledge.

8.4. Recommendations

We would like to conclude this chapter with various recommendations, ranging from general tips to more specific advice. First of all, a piece of advice that holds in much more generality than simply the field of spike sorting: readers should *never* use methods that they do not understand. In the context of spike sorting, and for data analysis in neurophysiology in general, *an excellent way to understand a method is to program it*. Today, there are many generalistic environments – or "ecosystems" as they are increasingly called by programmers – for data analysis: Python[39], R[40]; these ecosystems provide a platform for the methods discussed in the literature to be rapidly and easily implemented. As an example, we invite the reader to refer to

[38] See: http://en.wikipedia.org/wiki/Bayesian_information_criterion.

[39] Official website: https://www.python.org/, with the additional packages Numpy, Scilab, Matplotlib (the Web site http://www.scipy.org/ can serve as an entry point).

[40] Official website: http://www.r-project.org/.

the analysis of the two datasets used in this chapter in R and Python[41]. This advice should not be understood to imply that for “serious” analysis authors should necessarily reprogram all methods for themselves; clearly, for algorithms such as *k*-means or EM for GMM, effective and *well-tested* code is available and should be preferred. Nevertheless, a good understanding of these two algorithms may be easily obtained even just by programming simple versions of them. The advantage of our recommended approach, which will prove massive in the medium or long term, is that it enables data analysts to unshackle themselves from the methods provided by manufacturers (generally amplifier manufacturers); in our experience, these methods are opaque and insufficiently adaptable[42]. After more than 15 years of working in spike sorting (among other things, thankfully), on various different species (rats, mice, monkeys, locusts, beetles, bees), various different tissue types (cerebellum, hippocampus, neocortex, antennal lobe, etc.) and with various types of electrode, we have learned that certain key stages: filtering, event detection, clustering methods must be adapted to suit the tissue type[43]. Once these adjustments have been made, it is very straightforward, in environments with the right support such as Python or R, to write a script with very few parameters (or even no parameters) that can perform the entire sorting process for a given dataset. With this is mind, we *strongly* recommend using a modernized version of the approaches used until the mid-1980s:

1) The use the first minute of recording (or the first few minutes) to estimate the templates (section 8.3.3) with GGobi followed by *k*-means or EM for a GMM if *k*-means does not produce satisfactory results, or *bagged clustering* [LEI 99] if both of these options are not satisfactory;

2) Establish a classification by template matching (section 8.3.3) or using filters (sections 8.3.8) *after resolving instances of superposition* (section 8.3.6), accounting for sampling jitter (section 8.3.10) and dynamic amplitude

[41] See the page dedicated to spike sorting on the author’s website: http://xtof.perso.math.cnrs.fr/sorting.html, examples of analysis may be found at the bottom of the page.

[42] These two problems prompted the author to first begin programming his own methods.

[43] This holds for species/tissue pairs; thus different methods are used for recordings in the antennal lobe (the insect equivalent of the olfactory bulb for vertebrates) for locusts and for beetles.

profiles[44] (sections 8.3.7 and 8.3.9). This classification should be based solely on a recording period that is "short" compared to the electrode drifting timescale, which triggers changes in the templates and should not be performed over the whole of the recording;

3) Correct the templates for drifting, if necessary, and establish a classification for the next period of recording.

In general, the use the median instead of the mean – this is particularly important for template estimation – and the median of the absolute value of the deviations with respect to the median instead of the standard deviation (in short, the *median absolute deviation*); these two estimators are "robust"[45]. These two recommendations are a lot more important than they might seem; in practice, they produce a considerable improvement in the reliability of the results, and not just for spike sorting. Finally, the literature on spike sorting, and on the analysis of neurophysiological data in general, is relatively opaque; the author firmly believes that this problem could be reduced, or perhaps completely solved, if users/developers gave *unrestricted access to their data and their programs*, or in other words if they conducted their research so as to be reproducible [STO 14, DEL 12]: this is the best way to achieve both individual and collective progress.

8.5. Bibliography

[ADR 22] ADRIAN E.D., FORBES A., "The all-or-nothing response of sensory nerve fibres", *Journal of Physiology*, vol. 56, no. 5, pp. 301–330, 1922.

[ANT 00] ANTIC S., WUSKELL J.P., LOEW L. *et al.*, "Functional profile of the giant metacerebral neuron of Helix aspersa: temporal and spatial dynamics of electrical activity in situ", *The Journal of Physiology*, vol. 527, no. 1, pp. 55–69, 2000.

[ASI 85] ASIMOV D., "The grand tour: a tool for viewing multidimensional data", *SIAM Journal on Scientific and Statistical Comput*ing, vol. 6, no. 1, pp. 128–143, 1985.

[BED 04] BEDARD C., KROGER H., DESTEXHE A., "Modeling extracellular field potentials and the frequency-filtering properties of extracellular space", *Biophysics Journal*, vol. 86, no. 3, pp. 1829–1842, 2004.

[BRE 09] BRÉMAUD P., *Initiation aux Probabilités et aux chaînes de Markov*, Springer, Berlin Heidelberg, 2009.

[44] We still need an effective algorithm for resolving superposition in the presence of dynamic amplitude profiles.

[45] See: http://en.wikipedia.org/wiki/Robust_statistics.

[BUZ 04] BUZSÁKI G., "Large-scale recording of neuronal ensembles", *Nature Neurosciences*, vol. 7, no. 5, pp. 446–451, 2004.

[CAL 73] CALVIN W.H., "Some simple spike separation techniques for simultaneously recorded neurons", *Electroencephalography and Clinical Neurophysiology*, vol. 34, no. 1, pp. 94–96, 1973.

[CAN 10] CANEPARI M., ZECEVIC D., *Membrane Potential Imaging in the Nervous System: Methods and Applications*, Springer, 2010.

[CHA 99] CHAPIN J., MOXON K., MARKOWITZ R. *et al.*, "Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex", *Nature Neuroscience*, vol. 2, no. 7, pp. 664–670, 1999.

[CHA 04] CHAPIN J., "Using multi-neuron population recordings for neural prosthetics", *Nature Neurosci*ence, vol. 7, no. 5, pp. 452–455, 2004.

[CLE 93] CLEVELAND W.S., *Visualizing Data*, Hobart Press, NJ, 1993.

[COO 07] COOK D., SWAYNE D.F., *Interactive and Dynamic Graphics for Data Analysis. With R and GGobi*, Springer, Springer Science+Business Media, LLC, New York, 2007.

[DEL 06] DELESCLUSE M., POUZAT C., "Efficient spike-sorting of multi-state neurons using inter-spike intervals information", *Journal of Neuroscience Methods*, vol. 150, no. 1, pp. 16–29, 2006.

[DEL 12] DELESCLUSE M., FRANCONVILLE R., JOUCLA S. *et al.*, "Making neurophysiological data analysis reproducible. Why and how?", *Journal of Physiology (Paris)*, vol. 106, nos. 3–4, pp. 159–170, 2012.

[DON 08] DONG Y., MIHALAS S., QIU F. *et al.*, "Synchrony and the binding problem in macaque visual cortex", *Journal of Vision*, vol. 8, no. 7, pp. 1–16, 2008.

[EYZ 55] EYZAGUIRRE C., KUFFLER S.W., "Further study of soma, dendrite, and axon excitation in single neurons", *Journal of General Physiol*ogy, vol. 39, no. 1, pp. 121–153, 1955.

[FAT 57] FATT P., "Electric potentials occurring around a neurone during its antidromic activation", *Journal of Neurophysiol*ogy, vol. 20, no. 1, pp. 27–60, 1957.

[GEO 86] GEORGOPOULOS A., SCHWARTZ A., KETTNER R., "Neuronal population coding of movement direction", *Science*, vol. 233, no. 4771, pp. 1416–1419, 1986.

[GER 64] GERSTEIN G.L., CLARK W.A., "Simultaneous studies of firing patterns in several neurons", *Science*, vol. 143, no. 3612, pp. 1325–1327, 1964.

[GLA 68] GLASER E., MARKS W., "On-line separation of interleaved neuronal pulse sequences", in ENSLEIN K., (ed.), *Data Acquisition and Processing in Biology and Medicine*, Pergamon, 1968.

[GLA 76] GLASER E.M., RUCHKIN D.S., *Principles of Neurobiological Signal Analysis*, Academic Press, New York, 1976.

[GOL 74] GOLDSTEIN S.S., RALL W., "Changes of action potential shape and velocity for changing core conductor geometry", *Biophysics Journal*, vol. 14, no. 10, pp. 731–757, 1974.

[GRA 95] GRAY C.M., MALDONADO P.E., WILSON M. *et al.*, "Tetrodes markedly improve the reliability and yield of multiple single-unit isolation from multi-unit recordings in cat striate cortex", *Journal of Neuroscience Methods*, vol. 63, nos. 1–2, pp. 43–54, 1995.

[GRO 70] GROVER F.S., BUCHWALD J.S., "Correlation of cell size with amplitude of background fast activity in specific brain nuclei", *Journal of Neurophysiology*, vol. 33, no. 1, pp. 160–171, 1970.

[HAR 32] HARTLINE H.K., GRAHAM C.H., "Nerve impulses from single receptors in the eye", *Journal of Cellular and Comparative Physiology*, vol. 1, no. 2, pp. 277–295, 1932.

[HAR 00] HARRIS K.D., HENZE D.A., CSICSVARI J. *et al.*, "Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements.", *Journal of Neurophysiology*, vol. 84, no. 1, pp. 401–414, 2000.

[HAS 09] HASTIE T., TIBSHIRANI R., FRIEDMAN J., *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed., Springer, 2009.

[HOD 52] HODGKIN A.L., HUXLEY A.F., "A quantitative description of membrane current and its application to conduction and excitation in nerve", *The Journal of Physiology*, vol. 117, no. 4, pp. 500–544, August 1952.

[HOM 09] HOMMA R., BAKER B.J., JIN L. *et al.*, "Wide-field and two-photon imaging of brain activity with voltage- and calcium-sensitive dyes", *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 364, no. 1529, pp. 2453–2467, 2009.

[LEI 99] LEISCH F., Bagged clustering, Working Paper no. 51, Vienna University of Economics and Business, 1999.

[LIN 14] LINDÉN H., HAGEN E., LESKI S. *et al.*, "LFPy: A tool for biophysical simulation of extracellular potentials generated by detailed model neurons", *Frontiers in Neuroinformatics*, vol. 7, no. 41, 2014.

[MAL 81] VON DER MALSBURG C., The correlation theory of brain function, Internal Report, MPI Biophysical Chemistry, available at http://cogprints.org/1380/01/vdM_correlation.pdf, nos. 81–82 1981.

[MCG 84] MCGILL K.C., DORFMAN L.J., "High-resolution alignment of sampled waveforms", *IEEE Transactions on Biomedical Engineering*, vol. 31, no. 6, pp. 462–468, 1984.

[MCN 83] MCNAUGHTON B.L., O'KEEFE J., BARNES C.A., "The stereotrode: A new technique for simultaneous isolation of several single units in the central nervous system from multiple unit records", *Journal of Neuroscience Methods*, vol. 8, no. 4, pp. 391–397, 1983.

[PLO 07] PLONSEY R., BARR R., *Bioelectricity: A Quantitative Approach*, Springer, Springer Science+Business Media, LLC, New York, 2007.

[POG 63] POGGIO G.F., MOUNTCASTLE V.B., "The functional properties of ventrobasal thalamic neurons studied in unanesthetized monkeys", *Journal of Neurophysiology*, vol. 26, pp. 775–806, 1963.

[POU 02] POUZAT C., MAZOR O., LAURENT G., "Using noise signature to optimize spike-sorting and to assess neuronal classification quality", *Journal of Neuroscience Methods*, vol. 122, no. 1, pp. 43–57, 2002.

[POU 05] POUZAT C., "Course 15 – Technique(s) for spike-sorting", *Les Houches*, vol. 80, pp. 729–785, 2005.

[POU 14] POUZAT C., DETORAKIS G.I., "SPySort: neuronal spike sorting with python", *Proceedings of the 7th European Conference on Python in Science (EuroSciPy'14)*, pp. 27–34, 2014.

[PRO 72] PROCHAZKA V., CONRAD B., SINDERMANN F., "A neuroelectric signal recognition system", *Electroencephalography and Clinical Neurophysiology*, vol. 32, no. 1, pp. 95–97, 1972.

[RAL 77] RALL W., "Core conductor theory and cable properties of neurons", *Handbook of Physiology, Cellular Biology of Neurons*, American Physiological Society, Bethesda, MD, 1977.

[ROB 75] ROBERTS W.M., HARTLINE D.K., "Separation of multi-unit nerve impulse trains by a multi-channel linear filter algorithm", *Brain Research*, vol. 94, no. 1, pp. 141–149, 1975.

[ROB 79] ROBERTS W.M., "Optimal recognition of neuronal waveforms", *Biological Cybernetics*, Springer, Berlin/Heidelberg, vol. 35, pp. 73–80, 1979.

[SIM 65] SIMON W., "The real-time sorting of neuro-electric action potentials in multiple unit studies", *Electroencephalography and Clinical Neurophysiology*, vol. 18, no. 2, pp. 192–195, 1965.

[STO 14] STODDEN V., LEISCH F., PENG R.D., *Implementing Reproducible Research*, Chapman & Hall/CRC The R Series/Taylor & Francis, Boca Raton, 2014.

[WES 00] WESSBERG J., STAMBAUGH C.R., KRALIK J.D. *et al.*, "Real-time prediction of hand trajectory by ensembles of cortical neurons in primates", *Nature*, vol. 408, no. 6810, pp. 361–365, 2000.

[WIL 99] WILLIAMS S.R., STUART G.J., "Mechanisms and consequences of action potential burst firing in rat neocortical pyramidal neurons", *The Journal of Physiology*, vol. 521, no. 2, pp. 467–482, December 1999.

[ZEC 89] ZECEVI D., WU J.Y., COHEN L.B. *et al.*, "Hundreds of neurons in the Aplysia abdominal ganglion are active during the gill-withdrawal reflex", *Journal of Neuroscience*, vol. 9, no. 10, pp. 3681–3689, 1989.