Network: Comput. Neural Syst. **9** (1998) R53–R78. Printed in the UK PII: S0954-898X(98)84153-5

TOPICAL REVIEW

A review of methods for spike sorting: the detection and classification of neural action potentials

Michael S Lewicki†

Howard Hughes Medical Institute, Computational Neurobiology Laboratory, The Salk Institute, 10010 N Torrey Pines Road, La Jolla, CA 92037, USA

Received 31 July 1998

Abstract. The detection of neural spike activity is a technical challenge that is a prerequisite for studying many types of brain function. Measuring the activity of individual neurons accurately can be difficult due to large amounts of background noise and the difficulty in distinguishing the action potentials of one neuron from those of others in the local area. This article reviews algorithms and methods for detecting and classifying action potentials, a problem commonly referred to as spike sorting. The article first discusses the challenges of measuring neural activity and the basic issues of signal detection and classification. It reviews and illustrates algorithms and techniques that have been applied to many of the problems in spike sorting and discusses the advantages and limitations of each and the applicability of these methods for different types of experimental demands. The article is written both for the physiologist wanting to use simple methods that will improve experimental yield and minimize the selection biases of traditional techniques and for those who want to apply or extend more sophisticated algorithms to meet new experimental challenges.

Contents

† E-mail: lewicki@salk.edu. Address after 1 January 1999: Computer Science Department and Center for the Neural Basis of Cognition, Carnegie Mellon University, 115 Mellon Institute, 4400 Fifth Avenue, Pittsburgh, PA 15213, USA.

0954-898X/98/040053+26$19.50

1. Introduction

The detection of neural spike activity is a technical challenge that is a prerequisite for studying many types of brain function. Most neurons in the brain communicate by firing action potentials. These brief voltage spikes can be recorded with a microelectrode, which can often pick up the signals of many neurons in a local region. Depending on the goals of the experiment, the neurophysiologist may wish to sort these signals by assigning particular spikes to putative neurons, and do this with some degree of reliability. In many cases, good single-unit activity can be obtained with a single microelectrode and a simple hardware threshold detector. Often, however, just measuring the activity of a single neuron is a challenge due to a high amount of background noise and because neurons in a local area often have action potentials of similar shape and size. Furthermore, simple approaches such as threshold detection can bias the experiment toward neurons that generate large action potentials. In many cases, the experimental yield can be greatly improved by the use of software spike-sorting algorithms. This article reviews methods that have been developed for this purpose.

One use of spike sorting is to aid the study of neural populations. In some cases, it is possible to measure population activity by using multiple electrodes that are spaced far enough apart so that each can function as a single independent electrode. Traditional methods can then be used, albeit somewhat tediously, to measure the spike activity on each channel. Automatic classification can greatly reduce the time taken to measure this activity and at the same time improve accuracy of those measurements.

A further advantage of spike sorting is that it is possible to study local populations of neurons that are too close to allow isolation by traditional techniques. If the activity of several neurons can be measured with a single electrode, it is possible with spike sorting to accurately measure the neural activity, even in cases when two or more neurons fire simultaneously. This capability is especially important for experimental investigations of neural codes that use spike timing.

The main message of this review is not that a single method stands out from others, but that there are several methods of varying complexity and the decision about which one is appropriate depends on the requirements of the experiment. The issues and methods discussed in this paper will be primarily from the viewpoint of electrical recording, but many of the problems apply equally well to other techniques of neural signal detection.

The outline of the review is as follows. We first illustrate the basic problems and issues involved in the reliable measurement of neural activity. Next we review several techniques for addressing these problems and discuss the advantages and limitations of each, with

respect to different types of experimental demands. We conclude by discussing some of the current directions of research in this area and the remaining challenges.

2. Measuring neural activity

The first link between neural communication and electrical signals was made by Luigi Galvani in 1791 when he showed that frog muscles could be stimulated by electricity. It was not until the 1920s, however, that the nerve impulses could be measured directly by the amplification of electrical signals recorded by microelectrodes. The basic electrical circuit is shown in figure 1. The circuit amplifies the the potential between the ground (usually measured by placing a wire under the scalp) and the tip of the microelectrode. The potential changes measured at the tip reflect current flow in the extracellular medium. Typically the largest component of this current is that generated by the action potential, but there can be many other, less prominent components. Signals that look much like cellular action potentials can be picked up from axonal fibre bundles, also called fibres of passage. These signals are typically much more localized and small than cellular action potentials, which can usually be tracked while the electrode is advanced many tens of microns. Another signal source is the field potential. This is typically seen in layered structures and results from the synchronous flow of current into a parallel set of dendrites. This signal is typically of sufficiently low bandwidth that it can be filtered out from the neural actional potentials.

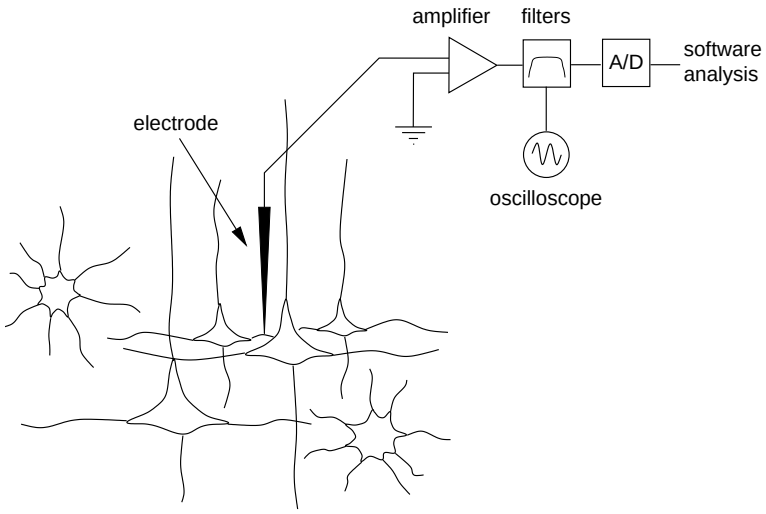


Figure 1. The basic set-up for measuring and analysing extracellular neural signals.

The shape of the electrode has some effect on what types of signals are measured. To some extent, the larger the tip of the electrode the greater the number of signals recorded. If the electrode tip is too large it will be impossible to isolate any one particular neuron. If the electrode tip is too small it might be difficult to detect any signal at all. Additionally, the configuration of the tip can be an important factor in determining what signals can be measured. Current in the extracellular space tends to flow in the interstitial space between cells and does not necessarily flow regularly. A glass electrode which has an O-shaped tip may pick up different signals than a glass-coated platinum–iridium electrode with a bullet-shaped tip. As is often the case in neurophysiology, what is best must be determined

empirically and even then is not necessarily reliable. For further discussions of issues related to electrical recording, see Lemon (1984).

The last step in the measurement is to amplify the electrical signal. A simple method of measuring the neural activity can be performed in hardware with a threshold detector, but with modern computers it is possible to analyse the waveform digitally and use algorithmic approaches to spike sorting (Gerstein and Clark 1964, Keehn 1966, Prochazka *et al* 1972). For a review on earlier efforts in this area, see Schmidt (1984). Previously software spike sorting involved considerable effort to set up and implement, but today the process is much more convenient. Several excellent software packages, many of which are publicly available, can do some of the more sophisticated analyses described here with minimal effort on the part of the user. Furthermore, the increased speed of modern computers makes it possible to use methods that in the past required prohibitive computational expense.

3. The basic problems in spike sorting

Many of the basic problems in spike sorting are illustrated in the extracellular waveform shown in figure 2. An immediate observation is that there are several different types of action potentials. Do these correspond to several different neurons? If so, how do we establish the correspondence? The waveform also shows a significant amount of background noise which could be from noise in the amplifier or smaller spikes from neurons in the local region. How do we reliably classify neurons in the presence of background noise? Another observation is that spikes from some cells overlap. How do we classify the overlapping action potentials? Methods to address these and other issues will be reviewed below. We will first start with simple algorithms that handle many situations and then in the next section discuss how to address the remaining issues with more sophisticated approaches.

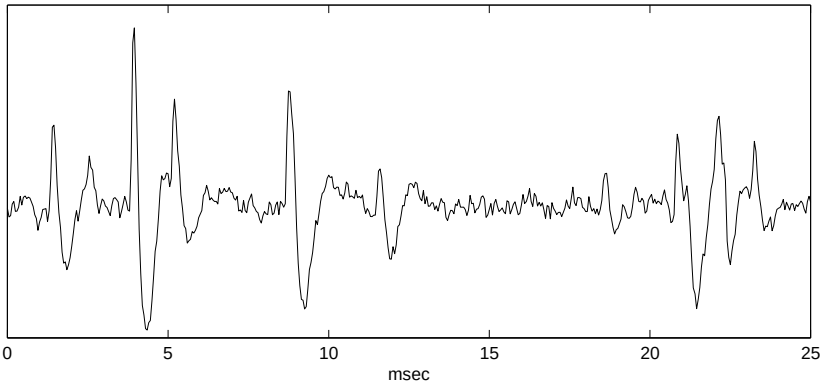


Figure 2. The extracellular waveform shows several different action potentials generated by an unknown number of neurons. The data were recorded from zebra finch forebrain nucleus LMAN with a glass-coated platinum–iridium electrode.

3.1. Threshold detection

Ideally, the experimenter would like to relate the spikes in the recorded waveform to the activity of a neuron in a local population. The reason this task is not hopeless is that neurons usually generate action potentials with a characteristic shape. This is not always true; for

example, some neurons generate several action potentials that change their shape over the course of the brief burst. This issue will be discussed in more detail below, but for now we will consider the case where the shape of the action potential is relatively stable.

For many neurons, the most prominent feature of the spike shape is its amplitude, or the height of the spike. One of the simplest ways to measure the activity of a neuron is with a voltage threshold trigger. The experimenter positions the recording electrode so that the spikes from the neuron of interest are maximally separated from the background activity. Neural activity is then measured with a hardware threshold trigger, which generates a pulse whenever the measured voltage crosses the threshold. This method is by far the most common for measuring neural activity. The obvious advantages of threshold detection are that it requires minimal hardware and software, and can often obtain exactly the information the experimenter wants. The disadvantage is that it is not always possible to achieve acceptable isolation.

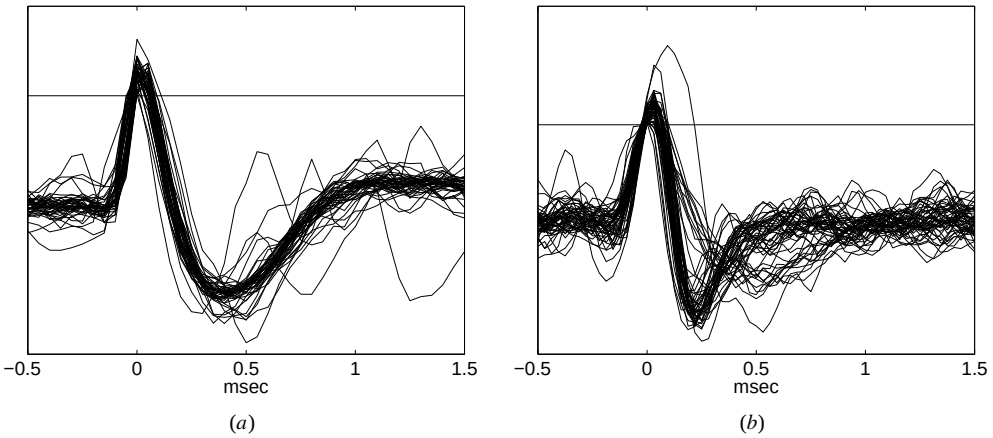


Figure 3. (*a*) A typical oscilloscope trace of a well isolated neuron. A trace is plotted every time the voltage crosses the threshold, indicated by the horizontal line. The portions of the waveforms preceding the trigger are also displayed. The data are from the same recording as the trace shown in figure 2. (*b*) A trace of a poorly isolated neuron. The data were recorded from zebra finch forebrain nucleus HVc with an parlyene-coated tungsten electrode.

The quality of isolation is often tested by looking at superimposed spikes on an oscilloscope. Figure 3 shows examples of a well isolated neuron and a poorly isolated neuron. In the well isolated case, there is still the presence of additional background spikes, but these have only a small effect on the quality of the isolation. In the poorly isolated case, two distinct spike shapes can be seen in the traces, and it is impossible to set the threshold so that one is isolated. Methods to handle the latter case will be discussed below, but first we consider issues involved in using a simple threshold.

A good method of checking the quality of the isolation over a longer period of time is with an *interspike interval histogram*. If the spike waveform in question is indeed a single unit, then there should be no interspike interval less than the refractory period, which for most cells is no less than 1 ms. The drawback of this method is that large numbers of spikes are required to be confident that there is indeed an isolated unit. Thus, this method is better suited for checking the quality of isolation over a long collection period. This is simple, however, because it is only necessary to store the interspike intervals.

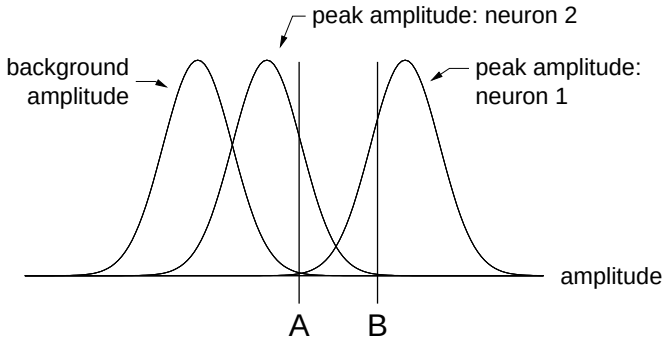


Figure 4. The figure illustrates the distribution of amplitudes for the background activity and the peak amplitudes of the spikes from two units. Amplitude is along the horizontal axis. Setting the threshold level to the position at A introduces a large number of spikes from unit 1. Increasing the threshold to B reduces the number of spikes that are misclassified, but at the expense of many missed spikes.

3.2. Types of detection errors

Very often it is not possible to separate the desired spikes from the background noise with perfect accuracy. The threshold level determines the trade-off between missed spikes (false negatives) and the number of background events that cross threshold (false positives), which is illustrated in figure 4. If the threshold is set to the level at A, all of the spikes from unit 1 are detected, but there is a very large number of false positives due the contamination of spikes from unit 2. If the threshold is increased to the level at B, only spikes from unit 1 are detected, but a large number fall below threshold. Ideally, the threshold should be set to optimize the desired ratio of false positives to false negatives. If the background noise level is small compared to the amplitude of the spikes and the amplitude distributions are well separated, then both of these errors will be close to zero and the position of the threshold hardly matters.

3.3. Misclassification error due to overlaps

In addition to the background noise, which, to first approximation, is Gaussian in nature (we will have more to say about that below), the spike height can vary greatly if there are other neurons in the local region that generate action potentials of significant size. If the peak of the desired unit and the dip of a background unit line up, a spike will be missed. This is illustrated in figure 5.

How frequently this will occur depends on the firing rates of the units involved. A rough estimate for the percentage of error due to overlaps can be calculated as follows. The percentage of missed spikes, like the one shown in figure 5(*b*), is determined by the probability that the peak of the isolated spike will occur during the negative phase of the background spike, which is expressed as

$$\%\text{missed spikes} = 100rd/1000 \tag{1}$$

where r is the firing rate in hertz and d is the duration of the negative phase in milliseconds. Thus if the background neuron is firing at 20 Hz and the duration of the negative phase is approximately 0.5 ms, then approximately 1% of the spikes will be missed. Note that this is only a problem when the negative phase of the background spikes is sufficiently large to cause the spikes of interest to drop below threshold.

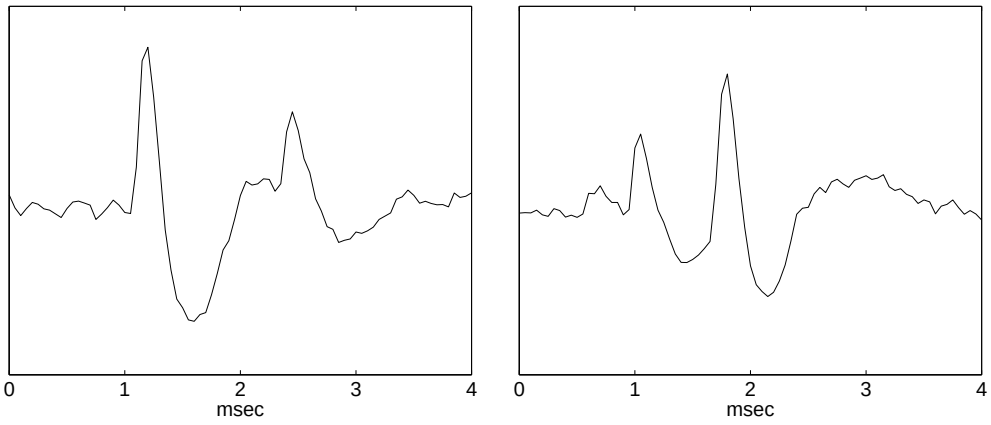


Figure 5. The peak level of the neuron of interest can change dramatically depending on the location and size of adjacent spikes.

Another potential source of error is when two background spikes combine to cross the trigger threshold. In this case, if two independent background neurons have rates r_1 and r_2, then their spikes will sum to cross threshold at a frequency of approximately $r_1 r_2 d/1000$, where d is the spike width in milliseconds. If the two background neurons have firing rates of 20 Hz and the spike width is 0.25 ms, then they will generate false positives at a rate of 0.1 Hz. A caveat with these estimation methods is that firing patterns of adjacent neurons can be highly correlated, so these equations will typically underestimate the frequency of missed or overlapping events.

Depending on experimental demands, these error rates may or may not be acceptable. Many neurophysiologists do not consider these situations as examples of a well isolated neuron, but having some knowledge of how these situations affect the accuracy of the isolation can help to improve yield if the experiment does not require perfect isolation. An additional caveat is that these rough estimates are based on the assumption that the neurons are independent and that the firing is Poisson in nature. Thus, the estimates of the number of overlaps are conservative and could be higher if neurons in a local region have correlated firing. Methods to handle these types of overlapping events will be discussed below.

4. Detecting and classifying multiple spike shapes

In the previous section, spike analysis was limited to detecting the height of the action potential. This is one of the simplest methods of analysis, but it is also very limited. In this section, we will review methods for detecting and classifying multiple spike shapes simultaneously. In this section, we will take the approach of starting with simple analysis methods and progress to more sophisticated approaches.

4.1. Feature analysis

The traces in figure 3(*b*) show two clear action potentials that have roughly the same height but are different in shape. If the shape could be characterized, we could use this information to classify each spike. How do we characterize the shape?

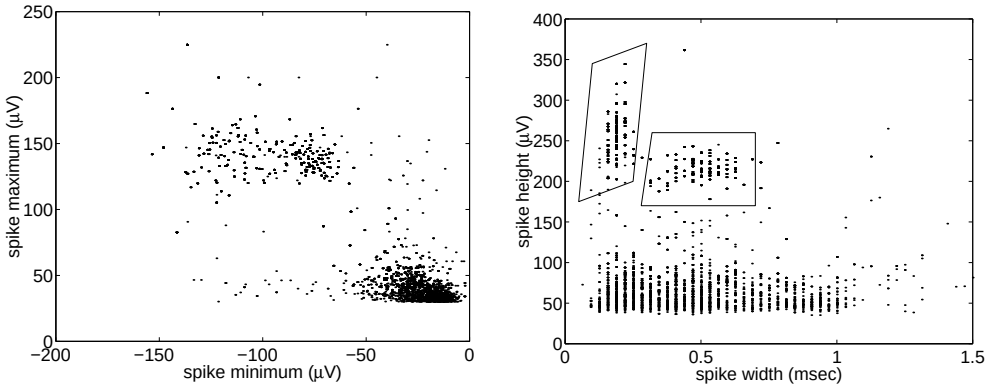


Figure 6. Different spike features can be used to classify different spike shapes. Each dot represents a single spike. Spikes with amplitude less than 30 μV were discarded. (*a*) A scatter plot of the maximum versus minimum spike amplitude. (*b*) A scatter plot of the spike height (the difference between the maximum and minimum amplitude) versus the spike width (the time lag between the maximum and minimum). The boxes show two potential cluster boundaries.

One approach is to measure features of the shape, such as spike height and width or peak-to-peak amplitude. This is one of the earliest approaches to spike sorting. It was common in these methods to put considerable effort into choosing the minimal set of features that yielded the best discrimination, because computer resources were very limited (Simon 1965, Feldman and Roberge 1971, Dinning 1981). For further discussions of feature analysis techniques, see Wheeler and Heetderks (1982) and Schmidt (1984).

In general, the more features we have, the better we will be able to distinguish different spike shapes. Figure 6(*a*) is a scatter plot of the maximum versus minimum spike amplitudes for each spike in the waveform used in figure 3(*b*). On this plot, there is a clear clustering of the two different spike shapes. The cluster positions indicate that the spikes have similar maximum amplitudes, but the minimum amplitudes fall into primarily two regions. The large cluster near the origin reflects both noise and the spikes of background neurons. It is also possible to measure different features, and somewhat better clustering is obtained with the spike height and width, as shown in figure 6(*b*). The vertical banding reflects the sampling frequency.

How do we sort the spikes? A common method is a technique called *cluster cutting*. In this approach, the user defines a boundary for a particular set of features. If a data point falls within the boundary, it is classified as belonging to that cluster; if it falls outside the boundary, it is discarded. Figure 6(*b*) shows an example of boundaries placed around the primary clusters. It should be evident that positioning of the boundaries for optimal classification can be quite difficult if the clusters are not distinct. There is also the same trade-off between false positives and missed spikes as there was for threshold detection, but now in two dimensions. Methods to position the cluster boundaries automatically will be discussed below.

In *off-line* analysis the cluster boundaries are determined after the data have been collected by looking at all (or a sample from) the data over the collection period. This allows the experiment to verify that the spike shapes were stable for the duration of the collection period. Clustering can also be performed *on-line* (i.e. while the data are being collected) if the clusters are stable. Methods for addressing unstable clusters will be discussed below.

4.2. Principal component analysis

Choosing features based on an intuitive idea of what might be useful is an *ad hoc* approach and, although simple, it can often yield poor cluster separation. Might there be a way of choosing the features automatically? One method for choosing features automatically is with *principal component analysis* (Glaser and Marks 1968, Glaser 1971, Gerstein *et al* 1983). The idea behind principal component analysis (PCA) is to find an ordered set of orthogonal basis vectors that capture the directions in the data of largest variation. The data are the original spikes from the recorded waveform. A sample from this data set is shown in figure 7(*a*). Each waveform is centred in the spike maximum to minimize the variability of the spike shapes.

To represent any particular data point (i.e. a spike) the principal components are scaled and added together. The scale factor for each component is sometimes called the *score*. The ith score is calculated by

$$s_i = \sum_t c_i(t)x(t) \tag{2}$$

where $x(t)$ is the spike and $c_i(t)$ is the ith principal component. Because the components

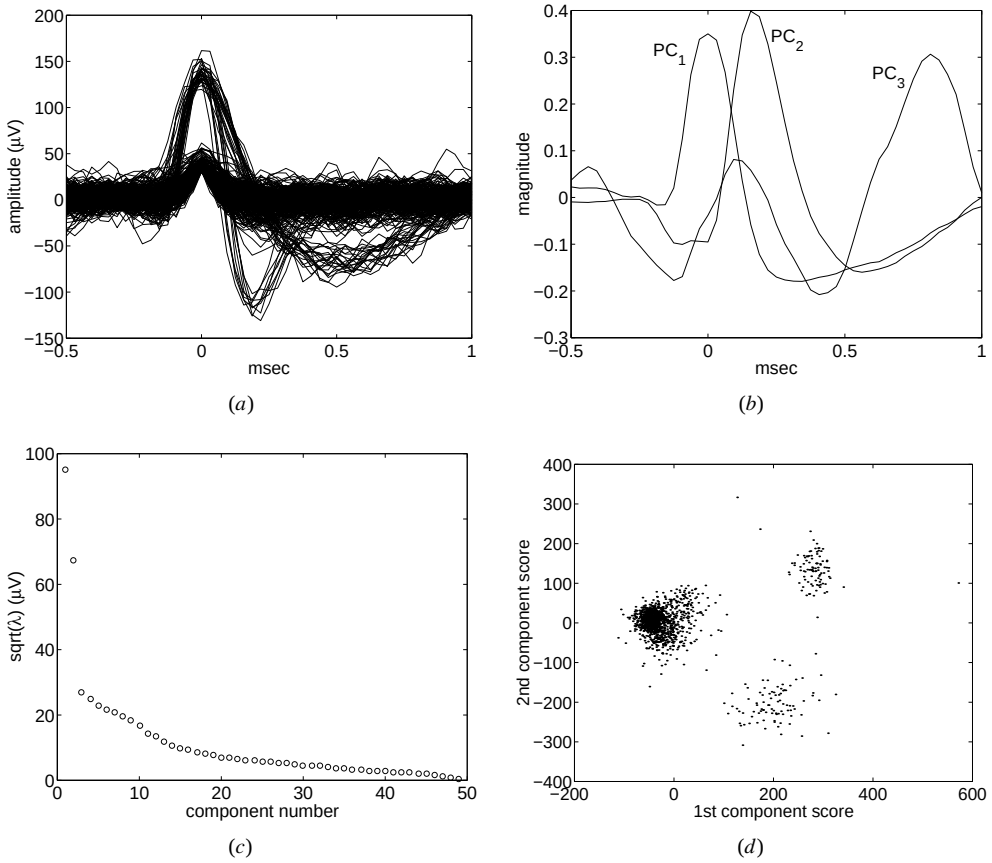


Figure 7. Results from principal component analysis of spike data. (*a*) The data used in the analysis. (*b*) The first three principal components. (*c*) The standard deviation of the scores in the direction of each component. (*d*) A scatter plot of the scores from the first two components.

are ordered in terms of how much variability they capture, adding together the first k components will describe the most variation in the data. Adding additional components yields progressively smaller corrections until the spike is described exactly. The principal component vectors are obtained by computing the eigenvectors of the covariance matrix of the data. In high-level, mathematical languages such as Matlab, Octave (a public domain program with similar functionality), or Mathematica, the principal components can be computed with just one line of code. For example, in the statistics toolbox in Matlab, once the waveforms have been placed in the rows of the matrix X, the line

```
[C,S,l]=princomp(X);
```

returns the principal components in the columns of matrix C, the scores of each waveform in the rows of matrix S, and the latent roots in the vector l.

The first three principal components for the spikes in figure 7(a) are shown in figure 7(b). The first component is the direction of largest variation in the data and has a generic spike-like shape. The second component also has a spike-like shape, but is offset with respect to the first component. The third component has no obvious interpretation and is the beginning of components that represent variability due to background noise.

Figure 7(c) plots the standard deviation of the scores in the direction of each component. The variances in these directions are sometimes called the latent roots (λ). The first three components account for 76% of the data variation ($100(\lambda_1 + \lambda_2 + \lambda_3)/\sum_i \lambda_i$); the first nine account for 90%. Only the first two have latent roots that are significantly above the background noise (about 30 μV), and thus using additional components for classification would yield little improvement in classification accuracy.

The scores of the first two components then can serve as features for classifying different spikes. Figure 7(d) shows a scatter plot of these two scores. Compared to the hand-chosen features in figure 6, it is evident that the first two components yield a clearer separation of the two spike shapes. One study comparing clustering methods found that principal components as features yield more accurate classification than other features, but are not as accurate as template matching (Wheeler and Heetderks 1982).

4.3. Cluster analysis

In the previous section, we described techniques for processing the data to reveal clusters that are relevant to classifying spike shapes. Although, the cluster boundaries can be set by hand, the question of how to set them automatically, or better yet optimally, is left open. *Cluster analysis* is one method for finding clusters in multidimensional data sets and classifying data based on those clusters.

A basic assumption underlying cluster methods is that the data result from several independent classes, each of which can be described by a relatively simple model. This assumption fits the case of spike sorting rather well, as each action potential arises from a different neuron. We have already seen examples of spike features that show clear clustering. The first task of clustering is to describe both the cluster location and the variability of the data around that location. The second task is, given a description of the clusters, to classify new data.

There are many methods for clustering (Duda and Hart 1973, Hartigan 1975, Everitt 1993). A simple approach, such as in nearest-neighbour or k-means clustering, is to define the cluster locations as the mean of data within that cluster. A spike is classified to whichever cluster has the closest mean, using Euclidean distance. This defines a set of implicit decision boundaries that separate the clusters. Figure 8 shows these boundaries for the same dataset

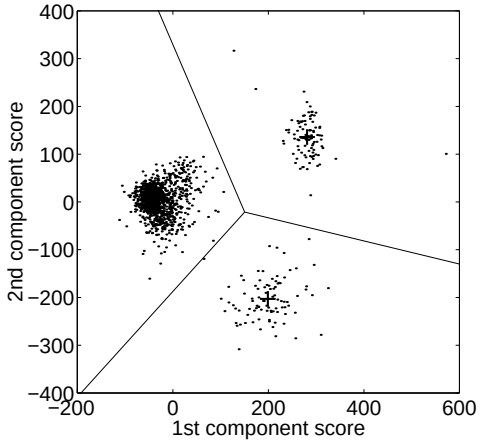


Figure 8. The lines indicate the decision boundaries for nearest-neighbour clustering. The cluster centres are indicated by the + symbols.

used in the previous section. Classifying in this manner uses only the information about the means and ignores the distribution of data within the cluster. This approach is adequate when the clusters are well separated, but breaks down when clusters significantly overlap or when the cluster shapes differ significantly from a spherical distribution.

4.4. Bayesian clustering and classification

Clustering can also be viewed as a model of the statistical distribution of the data. This approach offers many advantages over those described above, perhaps the biggest of which is that it quantifies the certainty with which spikes are classified. In this section, we briefly review the theory of this approach. For more thorough reviews of this topic, see Duda and Hart (1973), Bishop (1995), Ripley (1996) and Cheeseman and Stutz (1996).

A common probabilistic approach to clustering is to model each cluster with a multivariate Gaussian, centred on the cluster. The likelihood of the data given a particular class c_k is given by

$$p(x|c_k, \mu_k, \Sigma_k) \tag{3}$$

where x is the spike data vector and μ_k and Σ_k are the mean and covariance matrix for class c_k. The contours of the data likelihood of this model are multidimensional ellipses. The simpler methods described above can be viewed as a special case of Gaussian clustering where the clusters are described by spherically symmetric Gaussians.

The clustering model assumes that the data are selected independently from the underlying classes. The marginal likelihood, i.e. not conditioned on the classes, is computed by summing over the likelihood of the K classes

$$p(x|\theta_{1:K}) = \sum_{k=1}^{K} p(x|c_k, \theta_k)p(c_k) \tag{4}$$

where $\theta_{1:K}$ defines the parameters for all of the classes, $\theta_{1:K} = \{\mu_1, \Sigma_1, \ldots, \mu_K, \Sigma_K\}$. The term $p(c_k)$ is the probability of the kth class, with $\sum_k p(c_k) = 1$. In spike sorting, $p(c_k)$ corresponds to the relative firing frequencies.

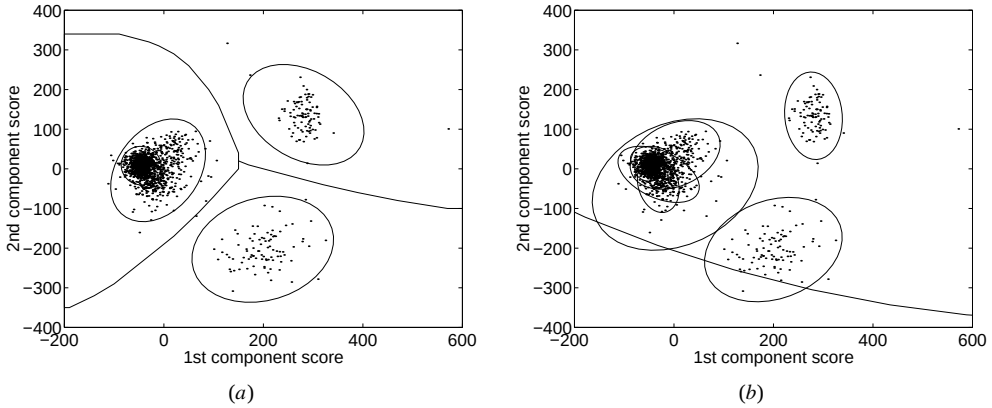


Figure 9. Application of Gaussian clustering to spike sorting. (*a*) The ellipses show the three-sigma error contours of the four clusters. The lines show the Bayesian decision boundaries separating the larger clusters. (*b*) The same data modelled with nine clusters. The elliptical line extending across the bottom is the three-sigma error contour of the largest cluster.

Classification is performed by calculating the probability that a data point belongs to each of the classes, which is obtained with Bayes' rule

$$p(c_k|x, \theta_{1:K}) = \frac{p(x|c_k, \theta_k)p(c_k)}{\sum_k p(x|c_k, \theta_k)p(c_k)}. \tag{5}$$

This implicitly defines the *Bayesian decision boundaries* for the model. Because the cluster membership is probabilistic, the cluster boundaries can be computed as a function of confidence level. This will yield better classification, because if the model is accurate the boundaries will be optimal, i.e. the fewest number of misclassifications.

The class parameters are optimized by maximizing the likelihood of the data

$$p(x_{1:\mathrm{N}}|\theta_{1:K}) = \prod_{n=1}^{N} p(x_n|c_k, \theta_{1:K})\,. \tag{6}$$

For the examples shown here, the cluster parameters were obtained using the publicly available software package AutoClass (Cheeseman and Stutz 1996). This package uses the Bayesian methods described above to determine both the means and the covariance matrices as well as the class probabilities, $p(c_k)$.

The ellipses (or circles) in figure 9 show the three-sigma (three standard deviations) error contours of the Gaussian model for each cluster. The figure illustrates two different models of the data, one with four clusters and one with nine. In this case, the clusters corresponding to the two large spikes appear in both solutions, but this illustrates that choosing the number of clusters is not always an easy task. This issue will be discussed further below. Note that the cluster in the middle, corresponding to the background spikes, is not modelled by a single class, but by two or more overlapping Gaussians.

The lines in figure 9(*a*) shows the Bayesian decision boundaries that separate the three larger clusters. The decision boundary for the smaller circular cluster is not shown, but it is in roughly the same position as the cluster's error contour.

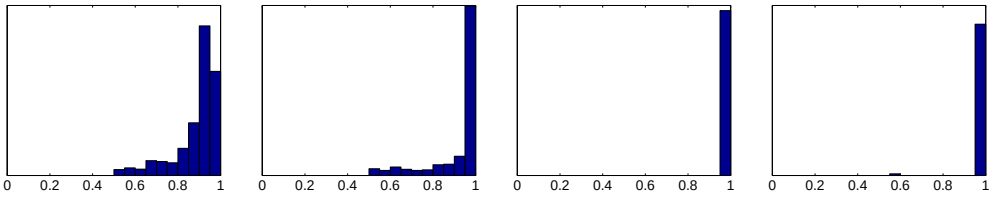
If the Gaussian cluster model were accurate, most of the data would fall within the three-sigma error boundary. In this case, three of the contours match the variability in the data, but the upper-right contour is significantly larger than the cluster itself. The reason

for this is due to the small number of *outliers* which correspond to overlapping spikes that cannot be well described by just the first two principal components.

4.4.1. Automatic removal of outliers One useful way of minimizing the effect of outliers and simultaneously improving the classification is to assign a large 'background' class with low cluster weight. This new class can be used to model the outliers, freeing up the other classes to capture the structure in the clusters themselves. Figure 9(*b*) shows a solution with a large background class having weight of 0.002. The three-sigma contour of this class is the large arc just below most of the data points. The decision boundaries for two clusters of interest are now elliptical and correspond approximately to the error contours. Points lying outside these boundaries are classified to the background class.

4.4.2. Bayesian classification An advantage of the Bayesian framework is that it is possible to quantify the certainty of the classification. This can often be a useful aid to the experimenter when making decisions about the isolation of spikes in different clusters. The probability of a spike being classified to a particular cluster is given by (5), which generates a probability for each cluster. By looking at the distribution of the probabilities for a class, it is possible to obtain some idea of how well separated that class is from others.

The histograms in figure 10 show the distribution of the probabilities for the most probable class across the data. In classes 3 and 4, nearly all of the data have probability equal 1.0, indicating that these points are assigned to their respective clusters with near certainty. For classes 1 and 2, only 26% and 68% of the points, respectively, have a class conditional probability greater than 0.95. This type of measure is particularly useful for monitoring the quality of the isolation during a prolonged period. A drop in isolation quality can indicate background noise or electrode drift.



(*a*) Class 1 (*b*) Class 2 (*c*) Class 3 (*d*) Class 4

Figure 10. Histograms of the probability of the most probable class for the data.

4.5. Clustering in higher dimensions and template matching

Although convenient for display purposes, there is no reason to restrict the cluster space to two dimensions; the algorithms also work in higher dimensions. Figure 11 shows the results of clustering the whole spike waveform. The waveforms are the class means and correspond to the average spike waveform for each class. By adding more dimensions to the clustering, more information is available which can lead to more accurate classification.

Using model spike shapes to classify new action potentials is also called *template matching* (Capowski 1976, Millecchia and McIntyre 1978). Earlier methods of template matching relied on the user to choose a small set of spikes that would serve as the templates. In clustering procedures, the spike templates are chosen automatically (Yang and Shamma 1988). If a Euclidean metric is used to calculate the distance to the template, then this

corresponds to nearest-neighbour clustering and assumes a spherical cluster around the template (D'Hollander and Orban 1979). The Bayesian version of classification by spike templates has the advantage that the classification takes into account the variation around the mean spike shape to give the most accurate decision boundaries.

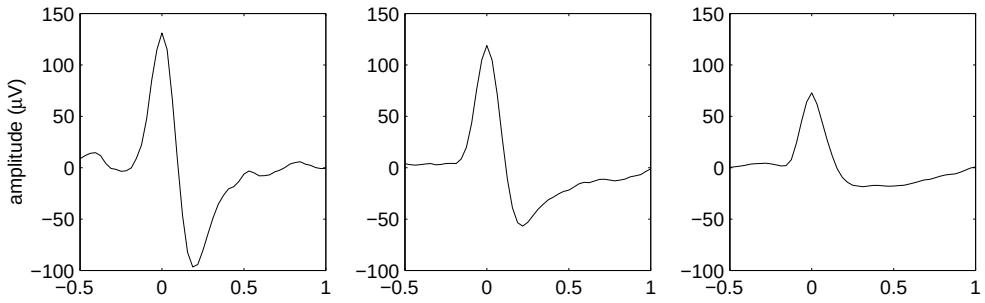


Figure 11. The waveforms show the cluster means resulting from clustering the raw spike waveforms.

4.6. Choosing the number of classes

One of the more difficult aspects of clustering approaches is choosing the number of classes. In Bayesian approaches, it is possible to estimate the probability of each model given the observed data (Jaynes 1979, Gull 1988, Cheeseman and Stutz 1988, Chickering and Heckerman 1997). If the assumptions of the model are accurate, this will give the relative probabilities of different numbers of classes. This approach was used by Lewicki (1994) in the case of spherical Gaussian mixture models. The software package AutoClass (Cheeseman and Stutz 1996), used in figures 9 and 11, estimates the relative probabilities for a general (multivariate) Gaussian mixture model. In figure 9, for example, the probability of the nine-class model was e^{160} times greater than the four-class model, suggesting overwhelming evidence in favour of the nine-class model. This procedure selects the most probable number of classes given the data and does not always favour models with more classes. In the same example, the probability of the nine-class model was e^{16} times greater than that of an eleven-class model. These numbers are calculated given the assumptions of the model and thus should be considered accordingly, i.e. they are accurate to the extent that the assumptions are valid.

Ideally, the number of classes would correspond to the number of neurons being observed, but there are several factors that prevent such a simple interpretation. The parameters of the classes are adapted to fit the distribution of the data. Individual neurons will not necessarily produce spikes that result in a distribution than can be well described by the model. In some special cases, such as with stationary spike shapes and uncorrelated noise, the clusters will be nearly spherical in which case the conclusions drawn from nearest-neighbour or symmetric Gaussian models can be very accurate. Many less ideal situations, such as correlated noise or non-stationary spike shapes, can still be accurately modelled by general Gaussian mixture models. In more complicated situations, such as neurons that generate complex bursts or when the background noise is non-stationary, the structure of the data can be very difficult to capture with a mixture model and thus it will be difficult both to predict the number of units and to make accurate classifications.

One approach to choosing the number of classes that avoids some of the assumptions

of the simple cluster models was suggested by Fee *et al* (1996a). The idea behind this approach is to use the interspike interval histogram to guide decisions about whether a class represents a single unit. When a neuron generates an irregularly shaped spike, such as a bursting neuron, many clustering algorithms often fit the resulting data with two or more classes. The method of Fee *et al* (1996a) groups multiple classes according to whether the interspike interval histogram of the group shows a significant number of spikes in the refractory period. Classification is done normally using the whole set of classes, but spikes classified as belonging to any of the classes in a group are labelled as the same unit. If overlapping action potentials are ignored, this approach can give biased results, because discarded overlaps could artificially create an interspike interval histogram that does not violate assumptions of the refractory period. It is important, then, with this approach, to calculate the interspike interval statistics in a region where overlaps can be accurately classified. Another potential drawback of this approach is that, because it relies on constructing interspike interval histograms, long collection periods may be required to achieve the desired degree of statistical certainty.

4.7. Estimating spike shapes with interpolation

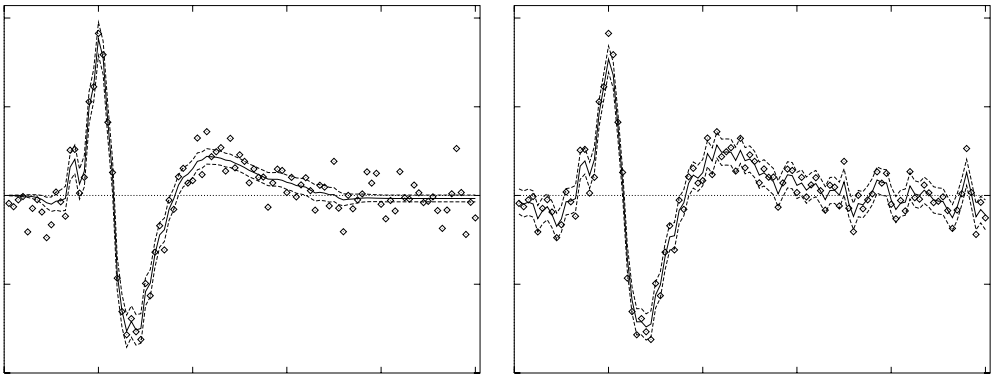
The clustering methods based on the raw waveform make no assumptions about the spike shape itself. From a Bayesian perspective, this is suboptimal, because it is known *a priori* that the spike shape will be smooth. What this means in practice is that many more data will be required to obtain a reasonable estimate of the spike shape. If the problem is viewed as one of *regularization*, there are several approaches that use prior knowledge and obtain more accurate estimates, with much fewer data. This issue is especially relevant in on-line clustering methods where it is important to obtain accurate spike classification based on as few spikes as possible.

The objective of regularization (also called noisy interpolation or smoothing) is to obtain an estimate of an underlying interpolant from noisy data. There is a large literature on this subject, and we give only a brief sketch of it here. For more detailed reviews, see Bishop (1995) and Ripley (1996).

When combined with the clustering approaches discussed in the previous section, regularization of the spike models places a constraint on the cluster mean so that it represents a smooth function. There is a trade-off between the smoothness of the interpolant and the amount of (additive) noise. If the interpolant is not regularized, it will tend to *overfit* the data. This is not a very big issue when there is a lot of data, but becomes more important when there are only a few examples. There are many approaches to determining the degree of smoothness. The algorithm of Lewicki (1994) uses a simple Bayesian method to determine both the background noise level and the smoothness of the spike functions. This is illustrated in figure 12(*a*) which shows that with very few data the estimated spike shape can be very noisy. This approach was extended further by MacKay and Takeuchi (1995) who used a model that allowed the smoothness to change over the time course of the interpolant (figure 12(*b*)). This type of model is particularly appropriate for describing action potentials, and the authors found that interpolants as accurate as standard regularization models could be obtained using only two thirds the amount of data.

4.8. Filter-based methods

Another approach to spike sorting uses the methods of optimal filtering (Roberts and Hartline 1975, Stein *et al* 1979, Andreassen *et al* 1979, Gozani and Miller 1994). The idea behind



(*a*) (*b*)

Figure 12. Estimating spike shapes with interpolation methods that use regularization allows more accurate estimates with fewer data. Each plot shows the raw data, the interpolant, and its error curves. (*a*) Bayesian interpolation with a single smoothness parameter for the whole waveform. (*b*) Bayesian interpolation using a smoothness that varies over the extent of the action potential. From MacKay and Takeuchi (1995).

this approach is to generate a set of filters that optimally discriminate a set of spikes from each other and from the background noise. This method assumes that both the noise power spectrum and the spike shapes can be estimated accurately. For each spike model, a filter is constructed that responds maximally to the spike shape of interest and minimally to the background noise, which may include the spikes of other units. The neural waveform is then convolved with the set of filters and spikes are classified according to which filter generates the largest response. This is analogous to the clustering methods above in which the metric used to calculate distance to the class mean is inversely related to the filter output. If the filters used to model the spike shapes were orthogonal, this would also be able to handle overlapping action potentials, but in practice this is rarely the case. Comparisons of spike classification on a common data set were carried out by Wheeler and Heetderks (1982) who found that optimal filtering methods did not classify as accurately as feature clustering using principal components or template matching.

5. Overlapping spikes

None of the methods described above explicitly deals with overlapping spikes. If two spikes are sufficiently separated in time, it is possible that the aforementioned methods will make the correct classification, but all of the methods reviewed thus far degrade severely when two spikes fire simultaneously. With the cluster cutting and Bayesian approaches to classification, it is possible to detect 'bad' overlaps as outliers. This gives the experimenter some gauge of the relative frequency of these events and whether they would compromise the results. There are many situations, however, where it would be desirable to both detect and accurately classify overlapping action potentials, e.g. investigations of local circuits or studies of spike-timing codes.

One simple approach to overlaps is to subtract a spike from the waveform once it is classified, in the hope that this will improve the classification of subsequent spikes. This approach requires a model of the spike shape (or template). It yields reasonable results

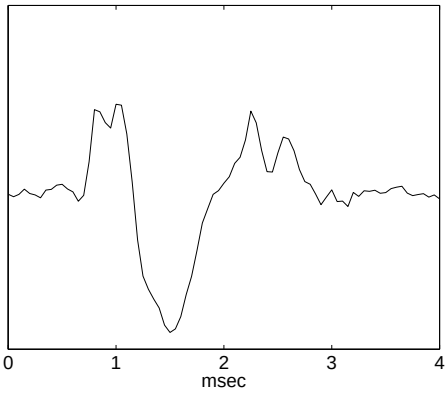


Figure 13. A high degree of overlap makes it difficult to identify the component spikes.

when two spikes are separated well enough so that the first can be accurately classified, but fails when the spikes are closer together like those shown in figure 13. Another problem with this approach is that the subtraction can introduce more noise in the waveform if the spike model is not accurate.

Another approach to the problem of overlaps is use neural networks to learn more general decision boundaries (Jansen 1990, Chandra and Optican 1997). Chandra and Optican (1997) reported that a trained neural network performed as well as a matched filter for classifying non-overlapping action potentials and showed improved performance for overlapping action potentials. A serious drawback of these approaches, however, is that the network must be trained using labelled spikes; thus the decision boundaries that are learned can only be as accurate as the initial labelling. Like the subtraction methods, these methods can only identify overlaps that have identifiable peaks.

One potential problem with subtraction-based approaches is that it is possible to introduce spurious spike-like shapes if the spike occurrence time is not accurately estimated. Typically, spike occurrence time is estimated to a resolution of one sample period, but often this is not sufficient to prevent artifacts in the residual waveform due to misalignment of the spike model with the measured waveform. The minimal precision of the time alignment can be surprisingly small, often a fraction of a sample period. Lewicki (1994) gave the following equation for error introduced by spike-time misalignment. For a given spike model, $s(t)$, the maximum error resulting from a misalignment of δ is

$$\epsilon = \delta \max_t \left| \frac{\mathrm{d}s(t)}{\mathrm{d}t} \right|. \tag{7}$$

For values of ϵ equal to the RMS noise level, typical δ's range from 0.1 to 0.5 sample periods. This equation gives a discrete number of time-alignment positions that must be checked in order to ensure an error less than ϵ.

A more robust approach for decomposing overlaps was proposed by Atiya (1992). When an overlap is detected, this approach compares all possible combinations of two spike models over a short range of spike occurrence times to find the combination with the highest likelihood. This approach can identify exact overlaps, but has the drawback of being computationally expensive, particularly for higher numbers of overlaps.

Lewicki (1994) introduced a computationally efficient overlap decomposition algorithm that addressed many of these problems. The idea behind the algorithm is to construct

a special data structure that can be used for classification, time alignment, and overlap decomposition, all at the same time and with minimal computational expense. The data structure is constructed from the spike shapes, once they are determined, and then used repeatedly for classification and decomposition when new spikes are detected.

In clustering and template-matching approaches, the largest computational expense is in comparing the observed spike to each of the clusters or templates. For single action potentials, this expense is insignificant, but if overlaps are considered then comparisons must be made for each spike combination (pair, triple, etc) and for all the alignments of the spikes with respect to each other. This quickly results in far too many possibilities to consider in a reasonable amount of time. It is possible, however, to avoid the expense of comparing the data from the observed spike to all possible cases. Using data structures and search techniques from computer science, it is possible to organize the cases in the data structure so that the spike combinations that most likely account for the data can be identified very quickly.

The algorithm of Lewicki (1994) used k-dimensional search trees (Bently 1975) to quickly search the large space of possible combinations of spike shapes that could account for a given spike waveform. k-dimensional trees are a multidimensional extension of binary trees, where k refers to the number of dimensions in the cluster means.

A simple method of constructing this type of search tree is to identify the dimension in the data with the largest variation and divide the data at the median of this dimension. This divides the data in two and determines the first division in the tree. This procedure is applied recursively to each subdivision until the tree extends down to the level of single data points. Each node in the tree stores what dimension is cut and the position of that cut. There are many algorithms for constructing these trees that use different strategies to ensure that the nodes in the tree are properly balanced (Ramasubramanian and Paliwal 1992).

To find the closest match in the tree to a new data point, the tree is traversed from top to bottom, comparing the appropriate dimensions at cutting points at each node. If each node divides on half of the subregion, then the closest match will be found in an average of $O(\log_2 N)$ comparisons, where N is the number of data points stored in the tree. In the case of spike sorting, these would correspond to the means of the models. To construct the tree requires $O(N \log_2 N)$ time, but once it is set up, each nearest-neighbour search is very fast. For purposes of calculating the relative probabilities of different clusters, it is not sufficient just to identify the closest cluster. For this calculation, it is necessary to find all the means within a certain radius (which is proportional to the background noise level) of the data point, and there are also algorithms that can perform this search using k-dimensional trees in $O(\log_2 N)$ time (Friedman *et al* 1977, Ramasubramanian and Paliwal 1992).

The overlap decomposition algorithm of Lewicki (1994) is illustrated in figure 14. The first step is to find a peak in the extracellular waveform. A region around the peak (indicated by the dashed lines) is selected. These data are classified with the k-dimensional tree which returns a list of spike sequence models and their relative probabilities. Each sequence model is a list of spikes (possibly only a single spike) and temporal positions relative to the waveform peak. The residual waveform (the raw waveform minus the model) of each remaining model is expanded until another peak is found. The relative probabilities of each sequence are recalculated using the entire waveform so far and the improbable sequences (e.g. probability < 0.001) are discarded. The cycle is repeated recursively, again using the k-dimensional tree to classify the waveform peak in each residual waveform. The algorithm terminates when no more peaks are found over the duration of the component spike models. The algorithm returns with list of spike model sequences and their relative probabilities. It is fast enough that it can be performed in real time with only modest computing requirements.

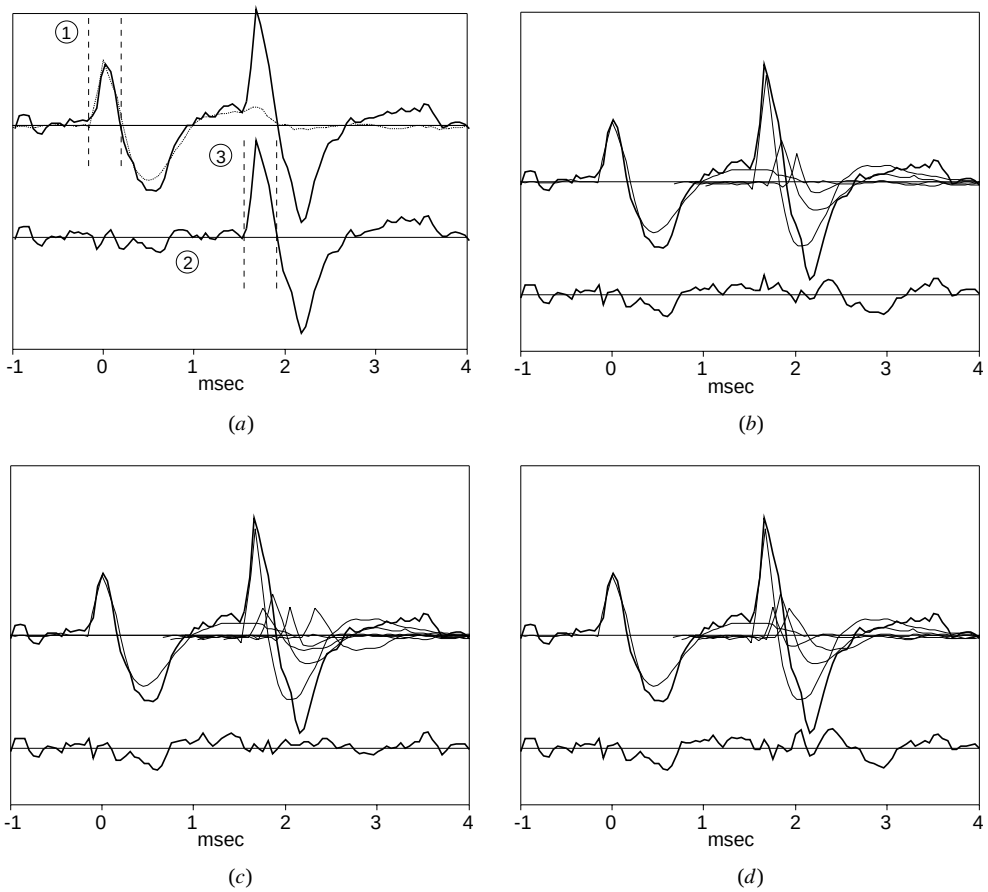


Figure 14. The overlap decomposition algorithm of Lewicki (1994). The thick lines indicate the raw waveform, the thin lines show the fitted spike models. The residual error (waveform minus model) is plotted below. (*a*) Step 1 of the algorithm identifies all plausible spike models in a peak region (indicated by the dashed lines). Step 2 expands each spike model sequence in the current list until another peak is found in the residual error. Step 3 calculates the likelihood of each spike model sequence and prunes from the sequence list all those that are improbable. The steps are repeated on the next peak region in the residual waveform for each sequence in the list. The algorithm returns a list of spike sequences and their relative probabilities. (*b*, *c*, *d*) Three different decompositions of the same waveform. The decomposition in (*b*) is twice as probable as that in (*c*) even though the latter fits the data better. The decomposition in (*d*) has small probability, because it does not fit the data sufficiently well.

One unforeseen consequence of a good decomposition algorithm is that there are actually many different ways to account for the same waveform. This is essentially the same *overfitting* problem that was mentioned in section 4.7: models with more degrees of freedom achieve better fits to the data, but can also result in less accurate predictions. This problem was addressed in Lewicki's algorithm by using Bayesian methods to determine which sequence is the most probable. For example, the four-spike sequence in figure 14(*b*) has twice the probability of the six-spike sequence in figure 14(*c*), even though the latter fits the data better (i.e. has less residual error). This approach does not make use of spike firing times, but in principle this information could also be incorporated into the calculation.

One limitation of this approach is that it assumes the clusters are spherical, which is equivalent to assuming a fixed spike shape with uncorrelated Gaussian noise. One approach for using this decomposition algorithm with non-spherical clusters is to use the approach of Fee *et al* (1996a) which uses several spherical clusters to approximate more general types of cluster shapes.

6. Multiple electrodes

There are many situations when two different neurons generate action potentials having very similar shapes in the recorded waveform. This happens when the neurons are similar in morphology and about equally distant from the recording electrode. One approach to circumventing this problem is to record from multiple electrodes in the same local area (McNaughton *et al* 1983). The idea is that if two recording electrodes are used, pairs of cells will be less likely to be equidistant from both electrodes (stereotrodes). This idea can be extended further to have four electrodes (tetrodes) that can provide four separate measurements of neural activity in the local area (McNaughton *et al* 1983, Recce and O'Keefe 1989, Wilson and McNaughton 1993). Under the assumption that the extracellular space is electrically homogeneous, four electrodes provide the minimal number necessary to identify the spatial position of a source based only on the relative spike amplitudes on different electrodes.

Having multiple recordings of the same unit from different physical locations allows additional information to be used for more accurate spike sorting. This can also reduce the problem of overlapping spikes: what appears as an overlap on one channel might be a isolated unit on another. Gray *et al* (1995) used recordings made with tetrodes in cat visual cortex to compare the performance of tetrodes with the best electrode pair and best single electrode. Using two-dimensional feature clustering, the tetrode recordings yielded an average of 5.4 isolated cells per site compared to 3.6 cells per site and 3.4 cells per site for the best electrode pair and best single electrode, respectively. Publicly available software for hand-clustering of tetrode data has been described by Rebrik *et al* (1997).

Many of the spike-sorting techniques developed for single electrodes extend naturally to multiple electrodes. Principal component analysis can also be performed on multiple-electrode channels to obtain useful features for clustering. In this case, the principal components describe the directions of maximum variation on all channels simultaneously. It is also possible to do clustering of all the channels simultaneously using the raw waveforms. Here each cluster mean represents the mean spike shape as it appears on each of the channels. Zhang *et al* (1997) have applied Bayesian clustering methods to tetrode data which show improved classification accuracy compared to two-dimensional clustering methods.

6.1. Independent component analysis

One recently developed technique that is applicable to multichannel spike sorting is *independent component analysis* (ICA), which is a method that was originally developed for doing blind source separation. The problem of blind source separation is illustrated in figure 15. The basic problem is to unmix N independent signals that have been linearly mixed onto N channels with unknown mixing weights. An assumption underlying this technique is that the unknown sources are independent. The objective is to learn the weights that best unmix the channels, i.e. transform the mixtures back into independent signals. Under this formulation, the signal separation is performed on a sample by sample

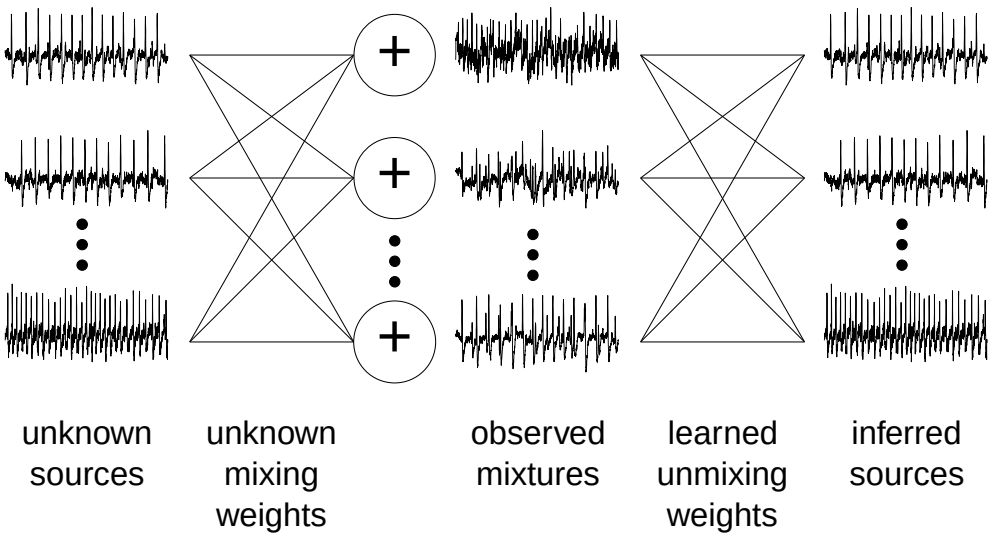


Figure 15. An illustration of blind source separation. *N* unknown sources are mixed linearly (with unknown mixing weights) to form *N* observed mixtures, one on each channel. Independent component analysis finds the unmixing weights that best transform the mixtures into independent signals.

basis, i.e. no information about spike shape is used. This is obviously a limitation, but could have advantages in cases where the underlying signals are not stationary.

ICA has been applied successfully to the analysis of EEG (Makeig *et al* 1997) and fMRI signals (McKeown *et al* 1998). Brown *et al* (1998) have applied ICA to the problem of separating channels of optical recordings of voltage-sensitive dyes in the sea slug *Tritonia*. An example of this analysis is shown in figure 16. Figure 16(*a*) shows short waveform sections from three channels of raw data. Note the presence of multiple neurons on channels 1 and 2, and the high amount of noise on channel 3. Figure 16(*b*) shows the inferred sources, after applying ICA to the whole dataset. Each source now contains mostly action potentials of similar shapes. Note that sources 1 and 2 can be combined (using different mixing proportions) to account for much of the structure in channels 1 and 2. Source 3 accounts for much of the variation in channel 3. Also note that much of the background noise on the channels has been removed. These variations are accounted for by other inferred sources (not shown).

One restrictive assumption of this approach is that the number of channels (electrodes) must equal the number of sources. This is obviously violated in the case of a single electrode but yields promising results for multichannel optical recordings. Another caution for this method is that it assumes that the sources are mixed linearly. Whether the technique is effective at separating the underlying signals using multiple electrodes has yet to be tested.

7. Related problems in spike sorting

In this section we describe several problems that can impair (sometimes severely) the usefulness of the methods described above. One of the more difficult aspects of spike sorting is making a method robust when the assumptions, both explicit and implicit, are violated. This lack of robustness is an important reason why many physiologists are reluctant to yield

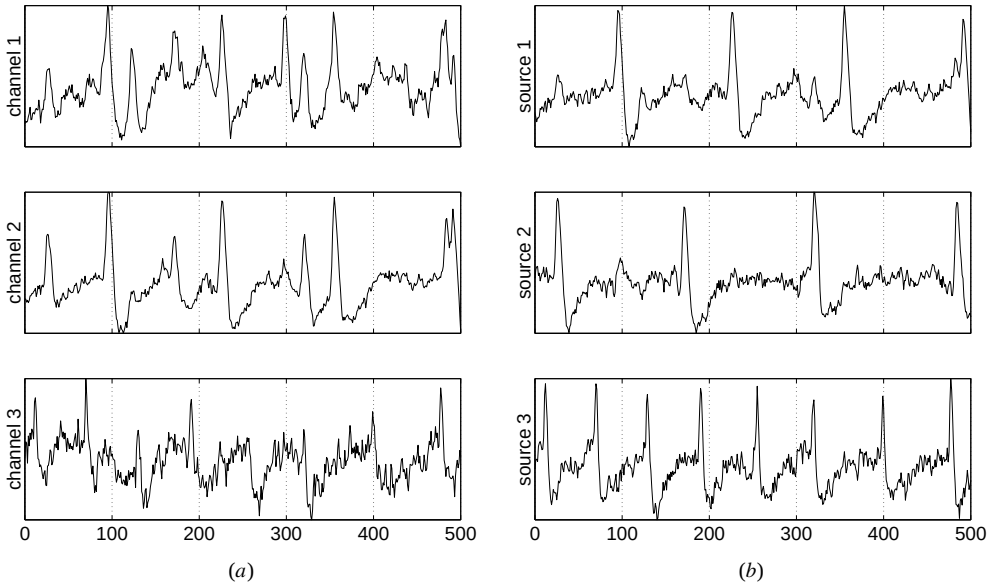


Figure 16. Independent component analysis (ICA) can be used to separate signals on multiple channels into separate sources. (*a*) Raw waveforms of an optical recording of voltage-sensitive dye in the sea slug *Tritonia*. Only three channels are shown. (*b*) Three of the independent sources derived by ICA. These can be scaled and combined to yield the raw traces. (Analysis and data courtesy of Dr Glen Brown, Salk Institute.)

careful hand isolation to often opaque software spike-sorting algorithms. Many algorithms work very well in the best case, when most assumptions are valid, but can fail badly in other cases. Unfortunately, it can be difficult to tell which circumstance one is in. Below we describe several common ways that the assumptions of many methods can be violated and how they have been addressed.

7.1. Burst-firing neurons

An implicit assumption in many of the methods above is that the spike shapes are stationary, i.e. they do not change with time. Many neurons, however, generate action potentials that have varying shape (Fee *et al* 1996b). Figure 17(*a*) shows an example where the action potential becomes progressively smaller. In clustering procedures this phenomenon results in a smearing or elongation of the cluster. Techniques such as multivariate Gaussian clustering or the cluster grouping method of Fee *et al* (1996a) can still classify bursts accurately, provided the attenuation is not too large and the individual spikes can still be detected. These approaches fail, however, when several neurons in a local group of neurons burst simultaneously. Such an example is shown in figure 17(*b*), which is recorded from the same site as figure 17(*a*). A different example of a burst is shown in figure 17(*c*). This shows that the attenuation of the spike height can be almost down to the noise level.

7.2. Electrode drift

Often during recording the electrode drifts slowly to a new position as the neural tissue settles in response to pressure from the advancement of the electrode. This results in a

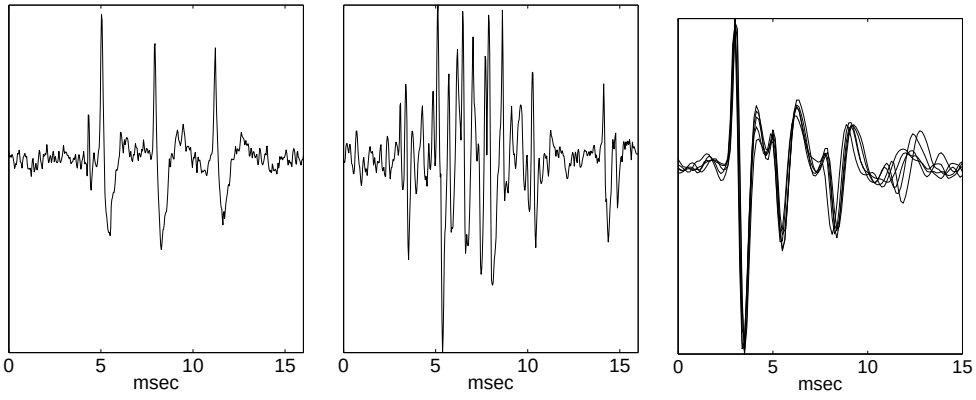


Figure 17. Two traces recorded from the same site in the zebra finch forebrain nucleus HVc. (*a*) Many neurons fire in short bursts in which the action potentials change shape. (*b*) In more complex bursts, the individual spikes are no longer visible. (*c*) The plot shows overlayed traces of different recordings of bursts from a ferret retinal ganglion cell (data courtesy of Chris Lee, Washington University, St Louis). The overall spike shapes in the bursts are more regular, but show much greater attenuation of the action potentials.

gradual change in the shapes of the action potentials. This problem can be addressed by the same methods that were used for bursting. Ideally, spike sorters would construct features or templates based on a limited period in the recording and allow these to change slowly over time, so as to compensate for electrode drift. Many of the clustering procedures have the advantage that once a set of classes is determined, the classes can be updated using new data with very little computational expense.

7.3. Non-stationary background noise

Another assumption that can be violated during the course of an experiment is the level of background noise. If the level of background noise remains constant, the classifications will be consistent throughout the trial. If the background noise level fluctuates, there will be many more misclassifications during high levels of noise. Methods that compute the certainty of the classification (e.g. using equation (5)) will be inaccurate when the background noise levels fluctuate. Ideally, the estimate of the reliability of the classification should vary with the background noise level, but this is seldom done because of the added complexity of implementing a time-varying noise model.

7.4. Spike alignment

As mentioned in the section on overlapping action potentials, the spike alignment can be an important factor in accurate classification (LeFever and DeLuca 1982, LeFever *et al* 1982, Lewicki 1994). This true for both template- and feature-based methods. Typically, spikes are aligned with respect to the amplitude peak, but this does not always yield accurate classification, particularly in cases when there are many samples during a spike peak. A simple approach to addressing this problem is to check adjacent alignments. If Bayesian clustering methods are used, different alignments are essentially different clusters and the relative probability of the different alignments can be calculated directly. This method was

used by Atiya (1992) and Lewicki (1994) and for overlap decomposition, where accurate spike alignment is especially important.

8. Summary

Which method is best? An early comparison of feature-based methods was done by Wheeler and Heetderks (1982) who concluded that template-matching methods yielded the best classification accuracy compared to spike-shape features, principal components, and optimal filters. Lewicki (1994) compared template-based, Bayesian clustering and classification to the commercial package Brainwaves, which relied on the user to define the two-dimensional cluster boundaries by hand. The methods gave similar results for well separated clusters, but the Bayesian methods were much more accurate for spike shapes that were similar. Template-based methods can fail for neurons that burst and can become increasingly inaccurate if there is electrode drift. The cluster grouping method of Fee *et al* (1996a) gives better classification in this situation compared to template-based methods. For overlapping action potentials, the method of Lewicki (1994) was shown to be nearly 100% accurate for action potentials that are significantly above the noise level.

Perhaps the most promising of recent methods for measuring the activity of neural populations is not an algorithm, but the recording technique of using multiple electrodes. Many of the difficult problems encountered with single-electrode recording vanish with multiple-electrode recordings. Gray *et al* (1995) showed that tetrodes yielded about two more isolated cells per site compared to the best single electrode, when the data were sorted with a simple two-dimensional clustering procedure. Bayesian clustering and classification shows promise to improve this yield even more (Fee *et al* 1996a, Sahani *et al* 1997, Zhang *et al* 1997).

A more practical question might be: what is the simplest method that satisfies experimental demands? For many researchers this is still a single electrode with threshold detection. Although simple, this technique can be time consuming and biased. Not only can neurophysiologists waste hours searching for well isolated cells, but in the end this search is biased towards cells that produce the largest action potentials which may not be representative of the entire population. Software spike sorting can reduce these biases, but this approach is still not in widespread use because of the difficulty in implementing even the simplest algorithms and also the added time required for obtaining more data. With modern computers and software this is no longer the case. If the raw waveform data can be transferred to the computer for software analysis, many of the algorithms described here can be implemented with simple programs using software packages such as Matlab, Octave, or Mathematica.

In the past several years, there has been much progress in spike sorting. It is now possible to replace much of the decision making and user interaction requirements of older methods with more accurate automated algorithms. There are still many problems that limit the robustness of many of the current methods. These include those discussed in section 7, such as non-stationary background noise, electrode drift and proper spike alignment. Possibly the most restrictive assumption of most methods is the assumption of stationary spike shapes. The method of Fee *et al* (1996a) addresses this problem to some extent, as can methods that use multiple electrodes, but currently there are no methods that can accurately classify highly overlapping groups of bursting action potentials. Decomposing overlapping action potentials with non-stationary shapes is largely an unsolved problem. Techniques that use multiple electrodes and incorporate both spike shape and spike timing information show promise in surmounting this problem.

Acknowledgments

The author thanks Glenn Brown, Mark Kvale, Chris Lee, Jamie Mazer and Kechan Zhang for help in preparing the manuscript.

References

Andreassen S, Stein R B and Oguztoreli M N 1979 Application of optimal multichannel filtering to simulated nerve signals *Biol. Cybern.* **32** 25–33

Atiya A F 1992 Recognition of multiunit neural signals *IEEE Trans. Biomed. Eng.* **39** 723–9

Bently J L 1975 Multidimensional binary search trees used for associative searching *Commun. ACM* **18** 509–17

Bishop C M 1995 *Neural Networks for Pattern Recognition* (Oxford: Clarendon)

Brown G D, Yamada S, Luebben H and Sejnowski T J 1998 Spike sorting and artifact rejection by independent component analysis of optical recordings from tritonia *Soc. Neurosci. Abstr.* **24** p 1670, no 655.3

Capowski J J 1976 The spike program: a computer system for analysis of neurophysiological action potentials *Computer Technology in Neuroscience* ed P P Brown (Washington DC: Hemisphere) pp 237–51

Chandra R and Optican L M 1997 Detection, classification, and superposition resolution of action-potentials in multiunit single-channel recordings by an online real-time neural-network *IEEE Trans. Biomed. Eng.* **44** 403–12

Cheeseman P and Stutz J 1988 AutoClass: A Bayesian classification system *Proc. 5th Int. Conf. on Machine Learning* (San Francisco: Morgan Kaufmann) pp 54–64

——1996 Bayesian classification (autoclass): theory and results *Advances in Knowledge Discovery and Data Mining* ed U M Fayyad *et al* (Menlo Park, CA: AAAI Press) pp 153–80, software available at `http://ic.arc.nasa.gov/ic/projects/bayes-group/autoclass`

Chickering D M and Heckerman D 1997 Efficient approximations for the marginal likelihood of Bayesian networks with hidden variables *Machine Learning* **29** 181–212

D'Hollander E H and Orban G A 1979 Spike recognition and on-line classification by an unsupervised learning system *IEEE Trans. Biomed. Eng.* **26** 279–84

Dinning G J 1981 Real-time classification of multiunit neural signals using reduced feature sets *IEEE Trans. Biomed. Eng.* **28** 804–12

Duda R O and Hart P E 1973 *Pattern Classification and Scene Analysis* (New York: Wiley)

Everitt B S 1993 *Cluster Analysis* (New York: Wiley) 3rd edn

Fee M S, Mitra P P and Kleinfeld D 1996a Automatic sorting of multiple-unit neuronal signals in the presence of anisotropic and non-gaussian variability *J. Neurosci. Meth.* **69** 175–88

——1996b Variability of extracellular spike wave-forms of cortical-neurons *J. Neurophysiol.* **76** 3823–33

Feldman J F and Roberge F A 1971 Computer detection and analysis of neuronal spike sequences *Inform.* **9** 185–97

Friedman J H, Bently J L and Finkel R A 1977 An algorithm for finding best matches in logarithmic expected time *ACM Trans. Math. Software* **3** 209–26

Gerstein G L, Bloom M J, Espinosa I E, Evanczuk S and Turner M R 1983 Design of a laboratory for multineuron studies *IEEE Trans. Systems, Man Cybern.* **13** 668–76

Gerstein G L and Clark W A 1964 Simultaneous studies of firing patterns in several neurons *Science* **143** 1325–7

Glaser E M 1971 Separation of neuronal activity by waveform analysis *Advances in Biomedical Engineering* vol 1, ed R M Kenedi (New York: Academic) pp 77–136

Glaser E M and Marks W B 1968 On-line separation of interleaved neuronal pulse sequences *Data Acquisition Process. Biol. Med.* **5** 137–56

Gozani S N and Miller J P 1994 Optimal discrimination and classification of neuronal action-potential wave-forms from multiunit, multichannel recordings using software-based linear filters *IEEE Trans. Biomed. Eng.* **41** 358–72

Gray C M, Maldonado P E, Wilson M and McNaughton B 1995 Tetrodes markedly improve the reliability and yield of multiple single-unit isolation from multiunit recordings in cat striate cortex *J. Neurosci. Meth.* **63** 43–54

Gull S F 1988 Bayesian inductive inference and maximum entropy *Maximum Entropy and Bayesian Methods in Science and Engineering, vol 1: Foundations* ed G J Erickson *et al* (Dordrecht: Kluwer)

Hartigan J A 1975 *Clustering Algorithms* (New York: Wiley)

Jansen R F 1990 The reconstruction of individual spike trains from extracellular multineuron recordings using a neural network emulation program *J. Neurosci. Meth.* **35** 203–13

Jaynes E T 1979 Book review of *Inference, method, and decision* by R D Rosenkrantz *J. Am. Stat. Assoc.* **74** 140

Keehn D G 1966 An iterative spike separation technique *IEEE Trans. Biomed. Eng.* **13** 19–28

LeFever R S and DeLuca C J 1982 A precedure for decomposing the myoelectric signal into its constituent action potentials. Part I: Technique, theory, and implementation *IEEE Trans. Biomed. Eng.* **29** 149–57

LeFever R S, Xenakis A P and DeLuca C J 1982 A precedure for decomposing the myoelectric signal into its constituent action potentials. Part II: Execution and test for accuracy *IEEE Trans. Biomed. Eng.* **29** 158–64

Lemon R 1984 *Methods for Neuronal Recording in Conscious Animals* (New York: Wiley)

Lewicki M S 1994 Bayesian modeling and classification of neural signals *Neural Comput.* **6** 1005–30, software available at `www.cnl.salk.edu/~lewicki/`

MacKay D J C and Takeuchi R 1995 Interpolation models with multiple hyperparameters *Maximum Entropy and Bayesian Methods, Cambridge 1994* ed J Skilling and S Sibisi (Dordrecht: Kluwer) pp 249–57

Makeig S, Jung T-P, Bell A J, Ghahremani D and Sejnowski T J 1997 Blind separation of auditory event-related brain responses into independent components *Proc. Natl Acad. Sci. USA* **94** 10979–84

McKeown M J, Jung T-P, Makeig S, Brown G, Kindermann S S, Lee T-W and Sejnowski T J 1998 Spatially independent activity patterns in functional magnetic resonance imaging data during the Stroop color-naming task *Proc. Natl Acad. Sci. USA* **95** 803–10

McNaughton B L, O'Keefe J and Barnes C A 1983 The stereotrode: a new technique for simultaneous isolation of several single units in the central nervous system from multiple unit records *J. Neurosci. Meth.* **8** 391–7

Millecchia R and McIntyre R 1978 Automatic nerve impulse identification and separation *Comput. Biomed. Res.* **11** 459–68

Prochazka V J, Conrad B and Sindermann F 1972 A neuroelectric signal recognition system *Electroenceph. Clin. Neurophysiol.* **32** 95–7

Ramasubramanian V and Paliwal K K 1992 Fast *k*-dimensional tree algorithms for nearest neighbour search with application to vector quantization encoding *IEEE Trans. Signal Proc.* **40** 518–31

Rebrik S, Tzonev S and Miller K 1997 Analysis of tetrode recordings in cat visual system *Computational Neuroscience: Trends in Research 1996* (New York: Plenum), software available at `mccoy.ucsf.edu`

Recce M L and O'Keefe J 1989 The tetrode: a new technique for multi-unit extracellular recording *Soc. Neurosci. Abstr.* **15** 1250

Ripley B D 1996 *Pattern Recognition and Neural Networks* (Cambridge: Cambridge University Press)

Roberts W M and Hartline D K 1975 Separation of multi-unit nerve impulse trains by a multi-channel linear filter *Brain Res.* **94** 141–9

Sahani M, Pezaris J S and Andersen R A 1997 Extracellular recording from adjacent neurons. I: a maximum-likelihood solution to the spike separation problem *Soc. Neurosci. Abstr.* **23** 1546

Schmidt E M 1984 Computer separation of multi-unit neuroelectric data: a review *J. Neurosci. Meth.* **12** 95–111

Simon W 1965 The real-time sorting of neuro-electric action potentials in multiple unit studies *Electroenceph. Clin. Neurophysiol.* **18** 192–5

Stein R B, Andreassen S and Oguztoreli M N 1979 Mathematical analysis of optimal multichannel filtering for nerve signals *Biol. Cybern.* **32** 19–24

Wheeler B C and Heetderks W J 1982 A comparison of techniques for classification of multiple neural signals *IEEE Trans. Biomed. Eng.* **29** 752–9

Wilson M A and McNaughton B L 1993 Dynamics of the hippocampal ensemble code for space *Science* **261** 1055–8

Yang X and Shamma S A 1988 A totally automated system for the detection and classification of neural spikes *IEEE Trans. Biomed. Eng.* **35** 806–16

Zhang K-C, Sejnowski T J and McNaughton B L 1997 Automatic separation of spike parameter clusters from tetrode recordings *Soc. Neurosci. Abstr.* **23** 504