Cluster method for analysis of transmitted information in multivariate neuronal data

M.-N. Chee-Orts, L. M. Optican

Laboratory of Sensorimotor Research, National Eye Institute, Building 10, Room 10C101, 9000 Rockville Pike, Bethesda, MD 20892, USA

Received: 3 August 1992/Accepted in revised form: 30 November 1992

Abstract. A new method for quantifying the transmitted information and channel capacity of high-dimensional data, based on cluster formation, is described. The method’s ability to handle high-dimensional data allows for a complete measurement of information transmitted by neuronal data. It is computationally efficient in terms of both processing time and memory storage. Application of the method to the responses of a VI neuron shows that more information was transmitted about the pattern of stimuli than about their color.

1 Introduction

Information theory serves as a convenient tool for the study of brain functions by providing a model-free method for quantifying the neuronal encoding and transmission of information. The two information measures commonly used in biological studies are transmitted information and channel capacity. Transmitted information measures the amount of information that the output of a channel gives about its input. It is a function of the probability distributions of both the input and output events, but is independent of the channel’s mechanisms. The channel capacity is the maximum transmitted information through the channel, achieved by optimizing the set of input probabilities.

In an earlier paper, Optican et al. (1991) developed a method for quantifying both these measures in neuronal codes by treating the neuron as a channel transmitting information about a stimulus in the temporal modulation of its response. Their method was based on the construction of a histogram, or contingency table, of the stimulus-response quantities, and will be referred to as the histogram method. It corrects for many of the biases, caused by the effects of quantization, small sample sizes and noise, that are known to plague historical attempts at applying information theory to small samples of noisy data (Miller 1955; Eckhorn and Pöpel 1974; Fagen 1978; Crowe et al. 1988).

However, the histogram method has two main limitations. These limitations arise when multidimensional codes are used to represent neuronal responses. For example, a temporal code could be based on three principal components of the responses, or on the 64 samples of the response waveforms taken over a 128 ms epoch. The first limitation is that the histogram method requires exponentially increasing computer memory storage with increase in dimensionality. Second, an accurate estimate of the $D$-dimensional histogram requires a rapid increase in sample size with increase in dimension, resulting in an exponential growth in the computer processing time. Consequently, the histogram method is unable to handle multivariate data of more than four dimensions. The limitation to low dimensions makes it difficult to measure the total amount of information carried by a neuron’s response. Optican et al. (1991) were able to partially overcome this problem by measuring the information transmitted by the first four principal components of the response waveform. Their approach, however, requires the assumption that most of the information is carried by the low components, an assumption that does not hold for all neurons in our recent studies.

We present a method, referred to as the cluster method, that performs efficient calculation of information with high-dimensional codes. The method surmounts both limitations of the histogram method by considering only regions of the response space that contain data observations. The cluster method is used to compute the transmitted information and channel capacity.

---

1 The principal components of a waveform are the complete orthonormal basis set that describe it with uncorrelated coefficients.
2 Memory storage $\sim N^D$, where $N$ is the number of bins in one dimension and $D$ is the total number of dimensions, respectively.
3 Computer processing time, $t \sim k^{2D}$, where $k \sim 2.855$, and $D$ is the dimension.
capacity for both simulated data and for the measured temporal firing patterns of a complex cell in the primate visual cortex (V1).

2 Method

2.1 Transmitted information

Transmitted or mutual information measures the amount of association between the input and output of a communication channel. It is given by (Abramson 1963)

\[ T(S; R) = \sum_i p(s_i) \sum_j \frac{p(r_j, s_i)}{p(s_i)} \log \frac{p(r_j, s_i)}{p(r_j)p(s_i)} \]  

(1)

where \( s_i \) and \( r_j \) are members of the input set \( S \) and the output set \( R \), respectively. The base 2 logarithmic function gives information in units of bits, where 1 bit of information is necessary and sufficient to discriminate between two equiprobable stimuli. If \( R \) and \( S \) are unrelated, \( p(r_j, s_i) = p(r_j)p(s_i) \), so that \( T(S; R) \) equals zero, i.e., no information is transmitted. Alternatively, if \( R \) and \( S \) are related, \( T(S; R) \) is a positive quantity. Given the probability terms, the channel capacity can be determined using an iterative scheme developed by Blahut (1988).

The histogram method explicitly constructs the stimulus-response probability terms that are needed in (1). The cluster method computes these probability values by considering the memberships of response vectors in equiprobable clusters that are formed by grouping the responses according to their proximity in the response space. A number of repetitions of the clustering process is required to ascertain that the calculated probabilities are representative of the data sample. If the transmitted information, \( T(S; R) \), is high, each cluster would contain responses that were elicited predominantly by a single stimulus, and there would exist a well-defined separation of the decision boundaries between the different response clusters. Conversely, a low \( T(S; R) \) would imply many different stimulus types within the same response cluster.

Originally, we attempted to implicitly determine these decision boundaries based on an extension of previous work on a neural-network model (Hertz et al. 1990). The input layer of the network consisted of the response clusters that were formed by the same procedure as in the cluster method. The transfer function at each cluster was a gaussian distribution function that transformed its member response vectors into vectors whose elements ranged between zero and one. For example, the transformed vector, \( \mathbf{v} \), had elements \( v_i = \exp(-K(x_i - m_i)^2) \), where \( x_i \) and \( m_i \) were the \( i \)th coordinates of the response vector and of the cluster centroid, respectively, and \( K \) was a constant. The purpose of using gaussian nodes was to assign weights to the vectors within a cluster based on their proximity to the centroid. The network’s outputs were the joint probabilities, \( p(r_j, s_i) \), obtained by using an exponential-type transfer function at the output nodes (Bridle 1989), rather than the traditional sigmoidal function. Under ideal conditions, the network learns the stimulus-response relationships directly from the training sets, without repetitive sampling and clustering of the response vectors (in contrast to the cluster method). However, for large and noisy data sets, it suffers from all the common ailments to which gradient-descent networks are susceptible, such as slow convergence and convergence to local minima. The explicit sampling procedures of the cluster method proved to be much more efficient at determining the probability terms.

2.2 Smoothed bootstrap sampling

It is often necessary to have a means of generating data whose properties reflect those of the original sample, because most experimental data is limited in quantity, and calculations based on them are usually subject to sample-size bias. A commonly used sampling scheme is the bootstrap method.

The standard bootstrap is a technique used for estimating various properties of a distribution \( F \) if the function itself is unknown, but a sample \( x_1, \ldots, x_n \) of observations from \( F \) is available (Efron 1979). For example, if one wishes to estimate some quantity \( g(F) \), the standard bootstrap approach is to simulate samples from the empirical distribution function \( \hat{F}_n \) by a uniform random selection with replacement from the observed data, \( x_1, \ldots, x_n \). However, this approach does not eliminate the sample size effects observed in the cluster method, because the generated data contains many replicated values. In fact, the standard bootstrap yields results with large variance, since regions in the response space may contain either a high density of response vectors or none at all, depending on the particular bootstrap repetition.

One can circumvent this problem by blurring the boundaries between regions which contain responses, so that the samples \( y_i \) are constructed from a smoothed version, \( \tilde{F}_i \), of \( F_i \). If \( x_{i,j} \) refers to data vector \( j \) from stimulus class \( i \), then its smoothed bootstrap value, \( y_{i,j} \), is given by (Silverman 1986)

\[ y_{i,j} = \tilde{x}_i + (x_{i,j} - \tilde{x}_i + \epsilon_i)/(1 + h^2)^{1/2} \]  

(2)

where \( \tilde{x}_i \) and \( \sigma^2_{x_i} \) are the ensemble mean and variance of data vectors in stimulus class \( i \) and \( \epsilon_i \) is a normal variate with zero mean and variance \( \sigma^2_{\epsilon_i} \). The parameter \( h \) allows one to adjust the degree of smoothing of \( \tilde{F}_i \). One can easily show that \( \tilde{F}_i = \tilde{x}_i \) and \( \sigma^2_{y_{i,j}} = \sigma^2_{x_i}/(1 + h^2) \).

In our calculations, a cloud of \( n \) \( y_{i,j} \) points was generated for each \( x_{i,j} \) point, where \( n \) was kept constant for all dimensions\(^4\). For our data sets, a reasonable value of \( n \) was 3. The calculated results were found to be relatively insensitive to the value of \( h \) used, provided that the latter was not so large as to cause a distortion.

---

\(^4\) We were able to overcome the requirement for sample size increase with dimension by considering only regions of the response space that contain data observations, and by using an empirical form for the information estimator (this is discussed in Sect. 2.3)
of the original distribution of points. In our calculations, \( h \) was chosen to be 0.05. Values of \( h \) that were larger than 0.2 tended to underestimate the information values of known data sets, especially for low dimensions.

2.3 Clustering of response space

In the present work, the stimuli used are discrete quantities, colored patterns chosen from a small set, but the responses are a continuous measure of the probability of an action potential occurring at that time in the response (Richmond et al. 1990). Each response datum is composed of 64 time samples, measured 2 ms apart, of the spike density function.

The response space is clustered by grouping the responses into \( D \)-dimensional hyperrectangles\(^5\), whose hypervolumes are given by \( 2^d \prod_{d=1}^D r^*_d \), where \( r^*_d \) is half the length of a side of the hyperrectangle in dimension \( d \), and its value depends on the data distribution in component \( d \) of the data set. Each \( r^*_d \) value has to be determined to perform the clustering process in \( D \) dimensions (the estimation of \( r^*_d \) is discussed in Sect. 2.6). Since the probability terms of (1) are computed by counting the number of response vectors in their respective clusters, it is important that the clusters occupy equal and non-intersecting volumes in the hyperspace. The clustering procedure used differs from other standard clustering techniques, such as the k-means algorithm (Jain and Dubes 1988), in that the number of clusters formed is not fixed, but depends on the cluster volume.

For each cluster \( i \), its centroid vector, \( m^i \), is given by

\[
\sum_{j=1}^{n_i} \frac{a_{ij}}{n_i} x^j ,
\]

where \( a_{ij} = 1 \) if vector \( x^j \) is a member of cluster \( i \) and zero otherwise; \( n_i \) is the number of members of cluster \( i \) \((n_i = \sum_k a_{ik})\); and \( N \) is the total number of vectors \((N = \sum_i n_i)\). Thus, \( a_{ij} = 1 \) if

\[
|m^i - x^j| < r^*_d , \quad \forall d
\]  

(3)

where \( d = 1, \ldots, D \) and \( D \) refers to the overall dimensionality of the cluster. The probability, \( p(j_i) \), is given by the number of response vectors elicited by stimulus \( j \), divided by the total number of vectors, \( N \). The probability, \( p(r, s) \), of finding a response vector in cluster \( i \) is given by \( n_i / N \). The joint probability, \( p(r, s) \), is given by the number of response vectors in cluster \( i \) elicited by stimulus \( j \), divided by the total number of vectors. These probability terms were averaged over many repetitions of the bootstrap and clustering process\(^6\).

2.4 Iterative clustering

The clustering procedure is carried out as follows. The first cluster is formed by using as its centroid a vector picked at random. Subsequent vectors either form new clusters, or are allocated to existing clusters, based on the following rules. For a vector, \( x \), if the centroid of

---

\(^5\) In our neuronal data, \( D = 64 \)

\(^6\) In our calculations, eight repetitions were usually sufficient

---

\(^7\) All distances in this paragraph refer to the euclidean distance
of $\gamma_D$ used in the cluster method was that which produced reasonably accurate $T_D$ values for data sets where the information values were known.

Two sets of simulated eight-component data; one containing uncorrelated pure signals in each component, and the other containing pure noise in each component, were used to establish the form for $\gamma_D$. In the first case, points corresponding to the 256 vertices of an eight-dimensional hypercube were used as the data. The criterion that had to be satisfied in this case was $T_D \sim \sum_{d=1}^D T_d$, i.e. the information in $D$ dimensions should be less than or equal to the sum of the information in each of its components, since no noise is present and the signals are uncorrelated. In the second case, gaussian noise was generated in each of the eight components of the data set. The criterion here was that $T_D \sim T_d$ for all $d = 1, \ldots, D$. This ensures that only the components which the signal is present.

In the histogram method, $\gamma$ was fixed at 2 and the number of sample points used for different numbers of dimensions was determined empirically from the neuronal data. The choice of $\gamma$ resulted in the histogram method overestimating the contribution of noisy components to the information.

2.6 Choice of $r^*$

When both stimulus and response are continuous and have normal distributions, it can be shown (Gallagher 1968) that

$$T(S; R) \sim \log(1 + C/r)$$

where $C$ and $r$ describe the distribution widths of the input stimulus and the output response, respectively. For our purpose, $C$ is a constant and $r$ corresponds to the radius of the cluster. As $r \to \infty$, all points become enclosed within one cluster, and $T(S; R)$ is appropriately zero, since no information is gained about any stimulus-response relationships. As $r \to 0$, the average number of points enclosed by each cluster decreases; the total number of cluster formed increases; and thus $T(S; R)$ increases. In a discrete and finite system, $\lim T(S; R)$ equals a large but finite value, because only a finite number of clusters can be formed. Since $T(S; R)$ varies with $r$, it is necessary to have a value of $r$, called $r^*$, that will yield a meaningful value of $T(S; R)$.

The linear dependence of $T(S; R)$ on $\log(1 + 1/r)$ is observed in our discrete system when $r$ is small compared to the variance of the data. The determination of $r^*$ will be illustrated with both simulated and neuronal data.

The six-component simulated data may be visualized as points scattered about the $2^6$ vertices of a six-dimensional trapezoid, with different lengths in the six dimensions. In this example, there were 64 stimuli, each with responses that corresponded to one of the 64 vertices of the trapezoid. If the points were strictly at the vertices, the information of the system would be $\log_2 64$ or 6 bits. However, by imposing six-dimensional gaussian noise on each point, where the signal to noise ratio is 6.79 dB\(^1\), the information transmitted by the system was lowered.

The dependence of $\langle T(S; R) \rangle$ on log $r$ is illustrated in Fig. 1a for the data in component one. The $\langle \rangle$ notation indicates that $\langle T(S; R) \rangle$ is a value obtained by averaging over a number of repetitions (eight in this case) of bootstrap sampling, clustering, bias correction, etc. Each error bar indicates one standard deviation of $\langle T(S; R) \rangle$. The standard deviation, $\sigma_x$ of $\langle X \rangle$ is given by $((X^2 - \langle X \rangle^2)(N - 1))^{1/2}$, where $N$ is equal to the number of repetitions. For small $r$ values, $\langle T(S; R) \rangle$ is linear in log $r$ as predicted by (5), and reaches a maximum value at some finite but small $r$ value. For a small range of intermediate $r$ values $\langle T(S; R) \rangle$ remains relatively flat, after which it undergoes a sudden drop in value, when most of the data are enclosed by a single cluster. The corrected transmitted information, $\langle \hat{T}(S; R) \rangle$, is given by the thick line. It can be seen that the two curves approach one another at the flat portion.

---

\(^8\) The system had stimuli labeled from 1 to 256, and independent responses that corresponded to the $2^6$ vertex locations of the hypercube.

\(^9\) It is one or zero, depending on whether the stimulus is on or off.

\(^10\) The signal to noise ratio was computed with the equation, $10 \log_{10}(\sigma_s^2/\sigma_n^2)$, where $\sigma_s^2$ and $\sigma_n^2$ are the variances of the signal and noise, respectively.
of the \( \langle T(S; R) \rangle \) curve. The value of \( r^* \) was chosen to be the smallest value of \( r \) at which the \( \langle T(S; R) \rangle \) curve both flattens out and approaches the \( \langle \hat{T}(S; R) \rangle \) curve. In these examples, the criterion used for establishing the proximity of the curves to one another was that \( \langle T(S; R) \rangle - \langle \hat{T}(S; R) \rangle \leq 0.08 \). Figure 1b shows \( \langle T(S; R) \rangle \) and \( \langle \hat{T}(S; R) \rangle \) values plotted against \( r \) when the data contains pure Gaussian noise. Based on the above rules, \( r^* = 33 \) and \( \langle \hat{T}(S; R) \rangle \mid r = 0.01 \).

Figure 1c illustrates an instance of a fairly strong signal, and Fig. 1d, a weak signal, in different components of a neuron’s response. It is easy to locate \( r^* \) (7th time sample, at 14 ms, in the 64-element response vector) in Fig. 1c, but \( r^*_0 \) (6th time sample) in Fig. 1d is more difficult to locate. In such cases, the value of \( r^*_0 \) is chosen to be the smallest \( r \) value at which all of the data are enclosed by a single cluster. In the neuronal data, \( r^*_0 \) for time sample \( d \) was generally found to occur within one standard deviation of the mean of the response at that time sample.

3 Results

3.1 Simulated data

In this section, calculations with simulated data are illustrated. Figure 2a, b compares the \( \langle \hat{T}(S; R) \rangle \) values computed using the histogram and cluster methods for the six-dimensional data set mentioned in the preceding section. The open circles denote the values \( \hat{T}_x \), the transmitted information for each component, \( x \). These are close to unity since the responses within each component are from either one of two equiprobable stimuli. The plain line indicates the sum of the transmitted information from each component. It serves as an upper bound for the joint information, for example, \( \hat{T}_1 + \hat{T}_2 \geq \hat{T}_{1,2} \). The cluster method (Fig. 2a) outperforms the histogram method (Fig. 2b) at higher dimensions (greater than three), as the latter tends to underestimate density values at high dimensions due to having an insufficient number of data sample points (Silverman 1986; Optican et al. 1991). Furthermore, the histogram method is not able to compute the transmitted information in five or more dimensions.

In the cluster method, increasing the dimensionality of the system has very little effect on the central processing unit (CPU) time of the calculation. However, the CPU time increases exponentially with \( n \), the number of cloud points generated per data point. Most of the CPU time is used to evaluate \( r^*_0 \) for each of the \( d \) components of the data set. For example, in the results shown in Fig. 2a, 88% of the 7.6 min of CPU time was used to evaluate the cluster radii. The histogram method used a total of 5.2 min of computational time for the results shown in Fig. 2b, of which 63% was used to calculate the transmitted information in four dimensions. To speed up calculations on the neuronal data sets which contain many trials (~1000 trials per data set), \( n \) was set to 1 and 3 during the radii-determining steps and the joint information calculations, respectively. All the calculations described in this paper were performed on four processing nodes of a Silicon Graphics 4D/480S machine.

3.2 Neuronal data

The cluster and histogram methods were applied to data obtained from individual neurons in the visual cortex of a monkey trained to perform a discrimination task (McClurkin et al. 1992). A foveal cue, which was one of six colored or six black and white patterned squares, was turned off before the stimulus appeared. The monkey had to choose one of the three parfoveal stimuli based on their color or pattern. There were a total of 36 stimuli derived from combinations of the six colors and six patterns. About 800 trials per neuron were obtained for all combinations of the cues and stimuli.

To illustrate the use of the cluster method, some of the results obtained will be analyzed. Two sets of information values were computed for a typical complex cell in area V1 of the visual cortex. They are the transmitted information about the six colors and six patterns of the stimuli.

Figure 3a shows the amount of information transmitted about pattern over the time course of the neuronal response. Both the histogram and cluster methods were used here, and there was good agreement in the results obtained by the two methods. This calculation (local information) involved determining \( \langle \hat{T}(S; R) \rangle \) independently at each component (time sample) of the 64-dimensional response vector.

The information curves for pattern (thick line) and color (intermediate-width line), calculated by the cluster method, are shown in Fig. 3b. This calculation involved the use of increasingly high-dimensional clusters with the progression of time, and so the histogram method
In about 10% of our neuronal data calculations, a fall-off in the joint information was observed at high dimensions. It is likely that in those instances, the distribution of data points at high dimensions was very dispersed, so that many of the clusters formed contained very few points. Coupled with the overestimation of \( T_p \) at high dimensions (see Sect. 2.5), this would lead to an underestimation of the transmitted information. Despite the unphysical behavior of the joint information at high dimensions, we were still able to obtain an approximate value of the total amount of information transmitted by the neuronal response by truncating the joint value before it declined.

4 Discussion

The cluster method was developed to overcome some of the limitations that were encountered in the histogram method, the most obvious being its requirement for large amounts of computer memory when processing high-dimensional data, as well as exponentially increasing processing time with high dimensions. These limitations hinder one from gaining a complete understanding of the functional role of the neuron in information encoding, since most neuronal responses require multi-dimensional descriptions. The cluster method is able to surmount these dimension-related obstacles by considering only regions of the response space which contain data.

The ability of the cluster method to handle high-dimensional data enables one to compute the total amount of information transmitted by a neuron directly from its response waveform. In addition, joint profiles of the information growth, like the ones shown in Fig. 3b, provide useful insights, such as the onset and rate of information encoding for different stimuli or cells. 

Acknowledgements. We would like to thank Dr. John W. McClurkin of LSR/NFL for useful discussion, and for providing the neuronal data, and Dr. John Hertz of NORDITA in Copenhagen for discussions on neural nets.

References


3.3 Limitations

In the present method, the accuracy of the information calculation depends on the clustering of the data points.

\(^{11}\) Time equals 0 corresponds to 20 ms after the stimulus onset
Miller GA (1955) Note on the bias of information estimates. Inf Theory Psychol Probl Methods 11B:95-100