Information Distortion and Neural Coding

Tomáš Gedeon† Albert E. Parker†
Alexander G. Dimitrov‡*
†Department of Mathematical Sciences and
‡Center for Computational Biology
Montana State University
Bozeman MT 59717

October 2, 2001

Abstract

Our main interest is the question of how neural ensemble activity represents sensory stimuli. In this paper we discuss a new approach to characterizing neural coding schemes. It attempts to describe the specific stimulus parameters encoded in the neural ensemble activity and at the same time determines the nature of the neural symbols with which that information is encoded.

This recently developed approach for the analysis of neural coding [6, 8] minimizes an intrinsic information-theoretic cost function (the information distortion) to produce a simple approximation of a coding scheme, which can be refined as more data becomes available. We study this optimization problem. The admissible region is a direct product of simplices. We show that the optimal solution always occurs at a vertex of the admissible region. This allows us to reformulate the problem as a maximization problem on the set of vertices and develop a new algorithm, which, under mild conditions, always finds a local extremum. We compare the performance of the new algorithm to standard optimization schemes on synthetic cases and on physiological recordings from the cricket cercal sensory system.

1 Introduction

One of the steps toward understanding the neural basis of an animal's behavior is characterizing the code with which its nervous system represents information. All computations underlying an animal's behavioral decisions are carried out within the context of this code.

Tools from information theory can be used to achieve two goals towards characterizing the neural coding scheme of a simple sensory system [8]. First, the functioning of a neural system is modeled as a communication channel. Although this model is stochastic, in this context a

*This research partially supported by DMS-NSF grant 291222 (TG), NSF-DGE grant 9972824 (AEP) and NIH grant MH12159 (AGD).

coding scheme consists of classes of stimulus/response pairs which form a structure akin to a dictionary: each class consists of a stimulus set and a response set, which are synonymous. The classes themselves are almost independent, with few intersecting members. The number of distinguishable classes is related to the mutual information between stimulus and response.

Secondly, we find high quality approximations of such a coding scheme. To do this, the neural responses are quantized to a small reproduction set. This quantization is optimized to minimize an information-based distortion function. Fixing the size of the reproduction produces an approximation of the coding scheme described above. The approximation can be refined by increasing the size of the reproduction. For the model described above, there is a critical size, beyond which further refinements do not significantly decrease the distortion. We choose the optimal quantization at this size to represent the coding scheme.

The admissible region over which the optimization is performed is a direct product of simplices. We show that the optimal solution always occurs at a vertex of the admissible region. This allows us to reformulate the optimization problem as the maximization of a new cost function on the set of vertices. We then develop a new algorithm, which, under mild conditions, always finds a local extremum.

Lastly, We compare the performance of the vertex search algorithm to standard optimization schemes on synthetic cases and on physiological recordings from the cricket cercal sensory system.

2 Preliminaries

2.1 The neural code

Deciphering the neural code of a sensory system means determining the correspondence between neural activity patterns and sensory stimuli. This task can be reduced further to three related problems: determining the specific stimulus parameters encoded in the neural ensemble activity, determining the nature of the neural symbols with which that information is encoded, and finally, quantifying the correspondence between these stimulus parameters and neural symbols. If we model the coding problem as a correspondence between the elements of an input set X and an output set Y, these three tasks are: finding the spaces X and Y and the correspondence between them.

Common approaches to this problem include stimulus reconstruction [25] and the use of impoverished stimulus sets to characterize stimulus/response properties [13]. However, these methods often introduce multiple assumptions that may affect the character of the obtained solution. Some of these approaches start with an assumption about the relevant structures of the space Y (e.g., a single spike in the first-order stimulus reconstruction method, or the mean spike rate over a defined interval) and proceed by calculating the expected stimulus features that are correlated with these codewords. Other approaches make an assumption about the relevant stimulus features (the space X), such as moving bars and gratings when investigating parts of the visual cortex, and proceed to study the patterns of spikes that follow the presentation of these features.

Observe that any neural code must satisfy at least two conflicting demands. On the one hand, the organism must recognize the same natural object as identical in repeated

exposures. On this level the response of the organism needs to be *deterministic*. On the other hand, the neural code must deal with uncertainty introduced by both external and internal noise sources. Therefore the neural responses are by necessity *stochastic* on a fine scale. In this respect the functional issues that confront the early stages of any biological sensory system are similar to the issues encountered by communication engineers in their work of transmitting messages across noisy media. With this in mind, we model the input/output relationship present in a biological sensory system as an *optimal information channel* [27] as in Figure 1A. An information channel characterizes the relationship between two random variables: an input X and an output Y. The structure of the neural code, which is stochastic on a fine scale, but deterministic on a large scale, emerges naturally in the context of an information channel using information theory.

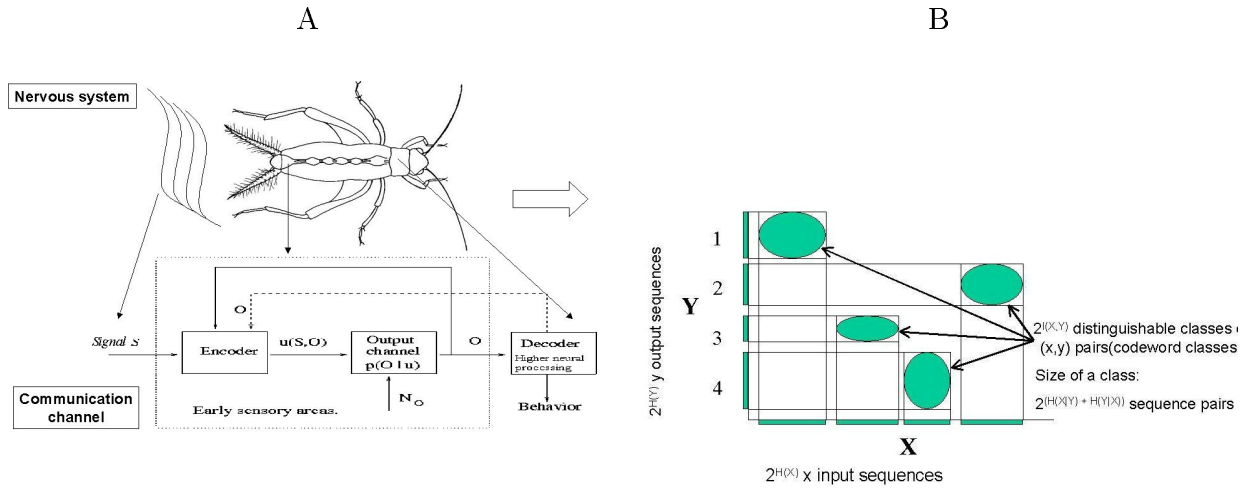

Figure 1: (A) The cricket cercal sensory system modeled as a communication channel. (B) The structure of a communication system. There are about $2^{nH(X)}$ stimulus (x) sequences, $2^{nH(Y)}$ response (y) sequences but only about $2^{nI(X,Y)}$ distinguishable equivalence classes y_{Ni} of (x, y) pairs.

2.2 Introduction to Information Theory

2.2.1 Basic Concepts

The basic object in information theory is an *information source* or a random variable X. X is a mathematical model for a physical system that produces a succession of symbols $\{x_1, x_2, \ldots, x_n\}$ in a manner which is unknown to us and is treated as random [5, 12].

The basic concepts of information theory are *entropy* and *mutual information*. The concept of entropy was first introduced in thermodynamics to provide a statement of the second law of thermodynamics [5]: the entropy of an isolated system is non-decreasing. In information theory, entropy is described as a measure of the uncertainty, or of the self information, of a random variable [5], and is defined as

$$H = -E_x \log p(x).$$

Next we define the *conditional* and *joint* entropy respectively as

$$\begin{aligned} H(Y|X) &= -E_{x,y} \log p(y|x) \\ H(X,Y) &= -E_{x,y} \log p(x,y). \end{aligned}$$

The notion of *mutual information* $I(X,Y)$ is introduced as a measure of the degree of dependence between a pair of random variables (X,Y):

$$\begin{aligned} I(X,Y) &= \log E_{x,y} \frac{p(x,y)}{p(x)p(y)} \\ &= H(X) + H(Y) - H(X,Y). \end{aligned}$$

Both entropy and mutual information are special cases of a more general quantity – the *Kullback-Leibler directed divergence* or *relative entropy* [19] between two probability measures on the same event space:

$$KL(p\|q) = E_p \log \left(\frac{p(x)}{q(x)} \right). \tag{1}$$

The information quantities H, I and KL depend only on the underlying probability distributions and not on the structure of X and Y. This allows us to evaluate them in cases where more traditional statistical measures (e.g. variance, correlations, etc.) simply do not exist.

Why are entropy and mutual information valid measures to use when analyzing an information channel between X and Y? Let $\{y_1, y_2, \ldots, y_n\}$ be i.i.d. observations from an information source Y. Then the Strong Law of Large Numbers provides theoretical justification for making inference about population parameters (e.g. response parameters) from data collected experimentally. In particular, the Shannon Entropy Theorem in this case assures that the entropy (and hence the mutual information) calculated from data taken experimentally converges to the true population entropy as the amount of data available increases. In the case of physiological recordings from a biological sensory system, $\{y_1, y_2, \ldots, y_n\}$ are not usually i.i.d.. For example, in the data that we present in this paper, we take a single, "long" recording of a neural response and break it up into observations of length 10ms. Inference made about population parameters from data collected this way is justified if we can assume that Y is ergodic. Now we may appeal to the Ergodic Theorem [3] and the Shannon-McMillan-Breiman Theorem [5] to justify the use of our information theoretic quantities.

2.2.2 Quantization Theory

A random variable Y can be related to another random variable Y_N through the process of *quantization* (lossy compression) [5, 12]. Y_N is referred to as the *reproduction* of Y. The process is defined by a map q from the probability space Y to Y_N, called a *quantizer*. In general, quantizers can be stochastic: q assigns to $y \in Y$ the probability that the response y belongs to an abstract class y_N. A deterministic quantizer is a special case in which q takes the values of 0 or 1 only. By Theorem A.4 in [8],

$$I(X, Y_{N+1}) \geq I(X, Y_N). \tag{2}$$

Furthermore, it can be shown that the mutual information $I(X, Y)$ is the least upper bound of $I(X, Y_N)$ over all possible reproductions Y_N of Y. Hence, the original mutual information can be approximated with arbitrary precision using carefully chosen reproduction spaces.

2.3 Neural systems as an information channels

Communication channels characterize a relation between two random variables: an input X and an output Y. When mapping this structure to neural systems, the output space is usually the set of activities of a group of neurons. The input space can be sensory stimuli from the environment or the set of activities of another group of neurons. We would like to recover the correspondence between stimuli and responses, which we call a *coding scheme* [29].

The early stages of neural sensory processing encode information about sensory stimuli into a representation that is common to the whole nervous system. We will consider this encoding process within a probabilistic framework [1, 18, 25]: *The input signal* X is produced by a source with a probability $p(x)$. This may be a sensory stimulus or the activity of a set of neurons. *The output signal* Y is produced by q with probability $p(y)$. This is the temporal pattern of activity across a set of cells. *The encoder* $Q(y|x)$ is a quantizer mapping X to Y. This will model the operations of a neuronal layer. In this framework we model a neuron or a group of neurons as a communication channel [5]. Results from information theory can be applied almost directly to this model for insights into the operation of a neural sensory system. Although the model is stochastic, an almost deterministic relation emerges naturally on the level of clusters of stimulus/response pairs. When restricted to codeword classes, the stimulus/response relation is almost bijective, as in Figure 1B. That is, with probability close to 1, elements of Y are assigned to elements of X in the same codeword class. We shall decode an output y as (any of) the inputs that belong to the same codeword class. Similarly, we shall consider the representation of an input x to be any of the outputs in the same codeword class.

2.4 Recovering a neural coding scheme

We recently developed a novel approach to finding a neural coding scheme through quantization of the neural response Y into a coarser representation in a smaller event space Y_N [8]. An important reason for using quantization for this purpose is the goal of using available data in the most efficient way. As pointed out in [15], the amount of data needed to support non-parametric estimates of coding schemes which contain long sequences of length T across N neurons grows exponentially with T and N. For some systems the required data recording time may well exceed the expected lifespan of the system. To resolve this issue we choose to sacrifice some detail in the description of the coding scheme in order to obtain robust estimates of a coarser description.

A quantization [5, 12] in this context is a stochastic map $q(y_N|y)$ of the neural representation Y into a coarser representation in a smaller event space Y_N. The random variables $X \to Y \to Y_N$ form a Markov chain. We characterize the quality of a quantization by a distortion function [5] and look for a minimum distortion quantization. The resulting relation between stimulus and reproduction, $q(y_N|x)$, will be a recovered approximation of

the neural coding scheme. By increasing the size of the reproduction, N, we can refine the approximation as much as the available data allows.

2.5 The distortion function

In engineering applications, the distortion function $D(\cdot,\cdot)$ is usually chosen in a fairly arbitrary fashion [5, 11],typically the Euclidean squared distance [26]. We want to avoid this arbitrariness. A quantization $q(y_N|y)$ produces a new random variable (a reproduction space) Y_N with associated probabilities $p(y_N)$. At the same time, quantization induces probabilities $p(x|y_N)$ which allow us to obtain a reconstruction of the input $\hat{p}(x) = \sum_N p(x|y_N)p(y_N)$ related to quantized observations $p(y_N)$. We view the distribution $p(x|y_N)$ as an approximation of the neural decoder $p(x|y)$. We require that this approximation is the best possible under the constraint that the number of classes N is fixed. The natural measure of the closeness of two distributions is the Kullback-Leibler divergence KL. For each fixed $y \in Y$ and $y_N \in Y_N$, $p(x|y)$ and $p(x|y_N)$ are a pair of distributions on the space X. We define our distortion function as the expected Kullback-Leibler divergence over all pairs (y, y_N)

$$D_I(Y, Y_N) = D_I(q(y_N|y)) := E_{y,y_N} KL(p(x|y_N)||p(x|y)).$$

Unlike the pointwise distortion functions usually investigated in information theory [5, 26], this one depends on the quantizer $q(y_N|y)$, through $p(x|y_N)$. We derive an alternate expression for D_I. Starting from the definition

$$\begin{aligned}
D_I &= \sum_{y,y_N} p(y, y_N)\ KL\big(p(x|y)\|p(x|y_N)\big) && (3)\\
&= \sum_{y,y_N} p(y, y_N) \sum_x p(x|y) \log \frac{p(x|y)}{p(x|y_N)} \\
&= \sum_{x,y,y_N} p(x, y, y_N) \Big(\log p(x|y) - \log p(x|y_N)\Big) && (4)\\
&= \sum_{x,y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} - \sum_{x,y_N} p(x, y_N) \log \frac{p(x, y_N)}{p(x)p(y_N)} && (5)\\
&= I(X, Y) - I(X, Y_N)
\end{aligned}$$

Step (4) uses the Markov property $p(x, y, y_N) = p(x|y)p(y, y_N)$. (5) is justified by using the identities $p(x, y) = \sum_{y_N} p(x, y, y_N)$ and $p(x, y_N) = \sum_y p(x, y, y_N)$, the Bayes property $p(x, y)/p(y) = p(x|y)$, and the fact that $\log p(x)$ is common for the two parts and cancels. This shows that the information distortion can be written as

$$D_I = I(X, Y) - I(X, Y_N)$$

as in [8]. This function can be interpreted as an *information distortion measure*, hence the symbol D_I. The only term in D_I that depends on the quantization is $I(X; Y_N)$, so we can replace D_I with the effective distortion

$$D_{eff} = I(X; Y_N)$$

in our optimization schemes. Our goal is to find a quantization $q(y_N|y)$ that minimizes the information distortion measure D_I for a fixed reproduction size N.

2.6 Finding the codebook

Following examples from rate distortion theory [5, 26], the problem of optimal quantization can be formulated as a maximum entropy problem [8, 14]. The reason is that, among all quantizers that satisfy a given set of constraints, the maximum entropy quantizer does not implicitly introduce additional constraints in the problem. In this framework, the minimum distortion problem is posed as a maximum quantization entropy problem with a distortion constraint:

$$\begin{array}{lll} \max\limits_{q(y_N|y)} H(Y_N|Y) & & \text{constrained by} \\ D_I(q(y_N|y)) \leq D_o & & \text{and} \\ \sum\limits_{y_N} q(y_N|y) = 1 \quad \text{and} & q(y_N|y) \geq 0 & \forall y \in Y \end{array} \tag{6}$$

The conditional entropy $H(Y_N|Y)$ and mutual information $I(X, Y_N)$ (the only term in D_I which depends on $q(y_N|y)$), can be written explicitly in terms of $q(y_N|y)$

$$\begin{aligned} H(Y_N \mid Y) &= E_{y,y_N} \log q(y_N|y) \\ &= \sum_{y,y_N} p(y)q(y_N|y) \log \left(q(y_N|y)\right) \end{aligned}$$

and

$$\begin{aligned} I(X, Y_N) &= \log E_{x,y_N} \frac{p(x, y_N)}{p(x)p(y_N)} \\ &= \sum_{x,y,y_N} q(y_N|y)p(x, y) \log \left(\frac{\sum_y q(y_N|y)p(x, y)}{p(x) \sum_y p(y)q(y_N|y)} \right) \end{aligned}$$

The optimal quantizer $q(y_N|y)$ induces a coding scheme from $X \to Y_N$ by $p(y_N|x) = \sum_y q(y_N|y)p(y|x)$ which is the most informative approximation of the original relation $p(x|y)$ for a fixed size N of the reproduction Y_N. Increasing N produces a refinement of the approximation, which is more informative (by (2)), so it has lower distortion and thus preserves more of the original mutual information $I(X, Y)$. The model of a coding scheme we use suggests that $D_I \propto -\log N$ for $N \leq N_c \approx 2^{I(X,Y)}$ and $D_I \approx \textit{constant}$ for $N \geq N_c$ [8]. Since we in general don't know $I(X, Y)$, we empirically choose N_c at which the rate of change of D_I with N decreases dramatically. This method allows us to study coarse but highly informative models of a coding scheme, and then to automatically refine them when more data becomes available.

3 Optimization schemes

The admissible region for the linear constraints in (6),

$$\Delta := \{q(y_N|y) \mid \sum_{y_N} q(y_N|y) = 1 \ \forall y \in Y \quad \text{and} \quad q(y_N|y) \geq 0\},$$

is a direct product of simplices. In section 5, we show that the optimal solution always occurs at a vertex of this region. We have devised three different algorithms to find this optimal solution. Two of these algorithms solve the system by starting in the interior of the feasible region and then using the method of *annealing* to find extrema. The third algorithm searches for extrema over Δ. When searching for the extrema of a general optimization problem, there is no known theory indicating whether using continuous, gradient-type algorithms is cheaper than searching over a finite, large set which contains the extrema. In this section, we investigate and compare these different approaches applied to (6).

3.1 Annealing

Using the method of Lagrange multipliers and D_{eff} instead of D_I we can reformulate the optimization problem as finding the maximum of the cost function

$$\begin{aligned} \max_{q(y_N|y)} F(q(y_N|y)) \quad &\equiv \quad \max_{q(y_N|y)} \Big(H(Y_N|Y) + \beta D_{eff}(q(y_N|y)) \Big) \\ \text{constrained by} \quad & \quad q(y_N|y) \in \Delta. \end{aligned} \tag{7}$$

This construction removes the nonlinear constraint from the problem and replaces it with a parametric search in $\beta = \beta(D_0)$. For small β the obvious optimal solution is the uniform solution $q(y_N|y) = 1/N$ [26]. It can be shown that as $\beta \to \infty$ solution of the problem (7) converges to a solution of the problem (6), which lies on the boundary of Δ. Therefore we need to track the optimal solution from $\beta = 0$ to $\beta = \infty$. We do this by incrementing β in small steps and use the optimal solution at one value of β as the initial condition for a subsequent β. To do this we must solve (7) at a fixed value of β. We have implemented two algorithms to solve this problem: an Augmented Lagrangian algorithm and an implicit solution algorithm.

3.1.1 Augmented Lagrangian

The Augmented Lagrangian algorithm is similar to other penalty methods in that the constraints to the problem are subtracted from F to create a new cost function to maximize

$$P(q, \mu) := F(q) - \frac{1}{2\mu} \sum_y (c_y(q))^2$$

where $c_y(q) := 1 - \sum_{y_N} q(y_N|y)$, the constraint imposed $\forall \ \ y \in Y$. The more infeasible the constraints $c_y(q)$ (when $1 - \sum_{y_N} q(y_N|y) >> 0$), the harsher the penalty in P.

The Augmented Lagrangian, however, avoids the ill-conditioning of other penalty methods (as $\mu \to \infty$) by introducing explicit approximations of the Lagrange multipliers into the

cost function at each optimization iteration. These approximations are constructed in such a way so that the solution to this algorithm satisfies the KKT conditions.

We use the Augmented Lagrangian, constructed specifically to deal with the equality constraints [23]

$$\mathcal{L}_A(q, \lambda, \mu) = \mathbf{F}(q) - \sum_y \lambda_y c_y(q) + \frac{1}{2\mu} \sum_y c_y(q)^2$$

and use a projected linesearch at each Augmented Lagrangian iteration to deal with the constraint $q(y_N|y) > 0$.

A Newton Conjugate Gradient method [23] is used to efficiently find a search direction for each linesearch. Once the active sets are identified, the theory assures us that this algorithm procures a stationary point (where $\nabla_q F = 0$) [17].

3.1.2 Implicit solution algorithm

This algorithm is based on the observation that extrema of F can be found by setting its derivatives with respect to the quantizer $q(y_N|y)$ to zero [8]. Solving this system produces the implicit equation (∇D_I depends on $q(y_N|y)$)

$$q(y_N|y) = \frac{e^{-\beta \frac{\nabla D_I}{p(y)}}}{\sum_{y_N} e^{-\beta \frac{\nabla D_I}{p(y)}}}. \tag{8}$$

Here ∇D_I denotes the gradient of D_I with respect to the quantizer. The expression (8) can be iterated for a fixed value of β to obtain a solution for the optimization problem, starting from a particular initial state.

3.2 Vertex search algorithm

Applying standard results from information theory [5], we have shown [8] that the function D_I is concave in $q(y_N|y)$. The domain Δ is a product of simplices and therefore convex. In section 5, we show that these two facts imply that the optimal solution of (6) always lies in a vertex of Δ (Corollary 13). Since the set of vertices is large, we implement a local search, linear in the order of the space Y, which leads, under modest assumptions, to a local maximum of (6) (Theorem 15). Details of the algorithm are given in (23).

4 Application to Data

As in [8], we now discuss the application of the method to synthetic data. Then we will turn our attention to the method applied to physiological recordings from the cricket cercal sensory system. In both of these scenarios, we compare the performance of the vertex search algorithm (23), the Augmented Lagrangian algorithm [23] with a projected Newton Conjugate Gradient line search and an implicit solution algorithm [8].

4.1 Synthetic Data

We analyze the performance of the three optimization schemes when using synthesized data (X, Y) drawn from the probability distribution shown in figure 2a. In this model we assume that X represents a range of possible stimulus properties and Y represents a range of possible spike train patterns. We have constructed four clusters of pairs in the stimulus/response space. Each cluster corresponds to a range of responses elicited by a range of stimuli. The mutual information between the two sequences is about 1.8 bits, which is comparable to the mutual information conveyed by single neurons about stimulus parameters in several unrelated biological sensory systems [7, 18, 24, 28]. For this analysis we assume the original relation between X and Y is known (the joint probability $p(x, y)$ is used explicitly).

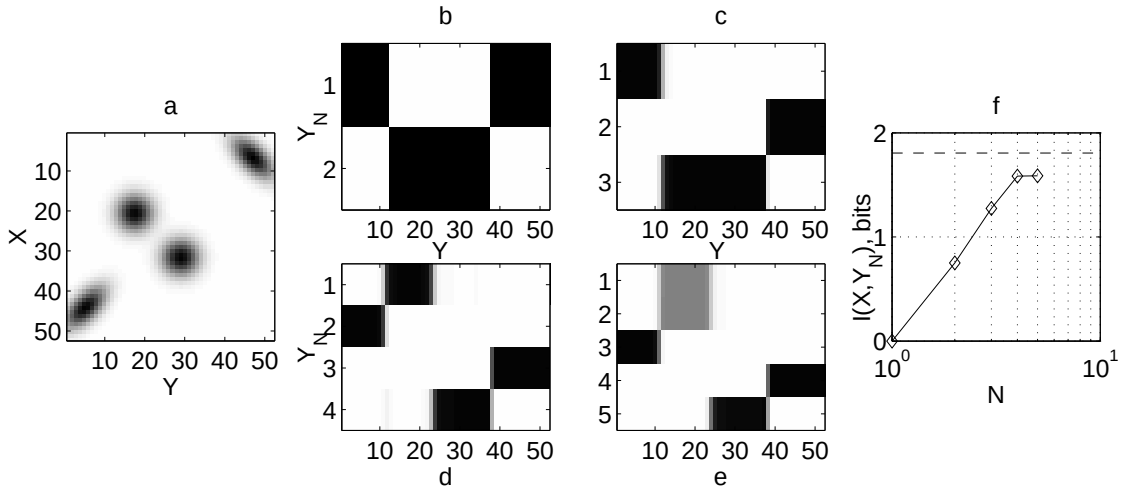

Figure 2: (a) A joint probability for the relation between two random variables X and Y, each with 52 elements. (b–e) The optimal quantizers $q(y_N|y)$ for $N = 2$, 3, 4 and 5 classes respectively. These panels represent the conditional probability $q(y_N|y)$ of a pattern y, a point on the horizontal axis in a, belonging to the class y_N, a point on the vertical axis in a. White represents $q(y_N|y) = 0$, black represents $q(y_N|y) = 1$, and intermediate values are represented by levels of gray. The behavior of the mutual information $I(X, Y_N)$ with increasing N can be seen in the log-linear plot (f). The dashed line is $I(X, Y)$, which is the least upper bound of $I(X, Y_N)$.

The optimal quantizer $q(y_N|y)$ for $N = 2$, 3, 4 and 5 is show in panels b–f of figure 2. The gray-scale map in these, and later representations of the quantizer, depicts $q(y_N|y) = 0$ with white, $q(y_N|y) = 1$ with black, and intermediate values with levels of gray. When an $N = 2$ class reproduction is forced as in panel (b), the algorithm recovers an incomplete representation of the coding scheme. The representation is improved for the $N = 3$ class refinement (c). The next refinement (d) with $N = 4$ separates all the classes correctly and recovers most of the mutual information. Further refinements (e) fail to split the classes and are effectively identical to (d). Note that classes $y_N = 1$ and 2 in (e) are almost evenly populated and the class membership there is close to a uniform 1/2. That is, $q(y_N = 1|y) \approx q(y_N = 2|y) \approx 1/2$ for $y : 12 \leq y \leq 23$. The quantized mutual information in (f) increases with the number of classes approximately as $\log N$ until it recovers about 90% of the original

mutual information (at $N = 4$), at which point it levels off.

A random permutation of the rows and columns of the joint probability in figure 2a has the same channel structure. The quantization is identical to the case presented in figure 2 after applying the inverse permutation and fully recovers the permuted classes (i.e., the quantization commutes with the action of the permutation group).

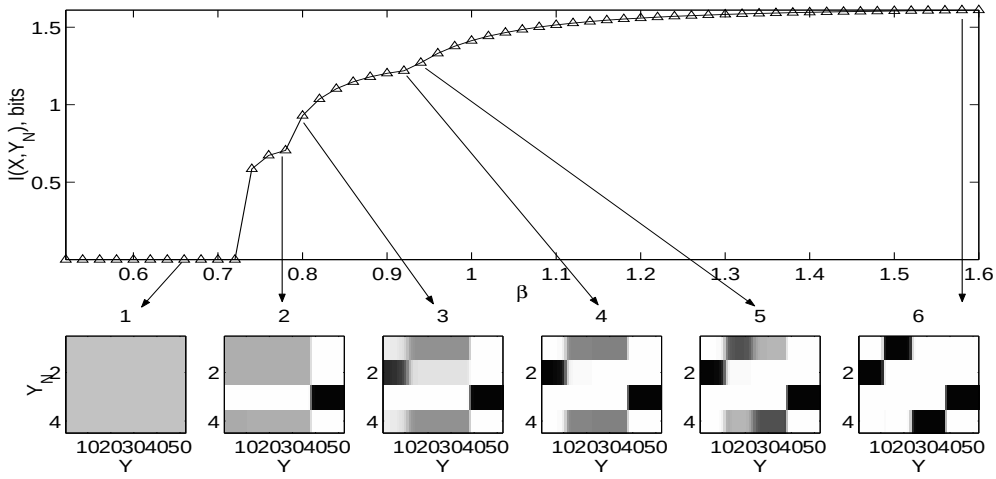

Figure 3: *For the data set in Figure 1A, the behavior of $D_{eff} = I(X, Y_N)$ (top) and the optimal quantizer $q(y_N|y)$ (bottom) as a function of the annealing parameter β.*

Further details of the course of the annealing optimization procedure (section 3.1) that lead to the optimal quantizer in panel (d) are presented in figure 3. The behavior of D_{eff} as a function of the annealing parameter β can be seen in the top panel. Snapshots of the optimal quantizers for different values of β are presented on the bottom row (panels 1 – 6). We can observe the bifurcations of the optimal solution (1 through 5) and the corresponding transitions of the effective distortion. The abrupt transitions ($1 \to 2$, $2 \to 3$) are similar to the ones described in [26] for a linear distortion function. We also observe transitions ($4 \to 5$) which appear to be smooth in D_{eff} even though the solution for the optimal quantizer seems to undergo a bifurcation.

Figure 4 gives a comparison of our optimization algorithms for this data set. For $N =$ 1, 2, 3 and 4, (A) shows the maximal mutual information procured by each algorithm, while (B) indicates the computational cost of each. The vertex search was the fastest and the Augmented Lagrangian the slowest of the three with an order of magnitude difference between each two algorithms. Each algorithm has its advantages, though, as the Augmented Lagrangian always gives a point that satisfies the KKT conditions and the vertex search does so under certain conditions (15). Although we do not have a complete theoretical understanding of the convergence of the implicit solution algorithm, it works very well in practice.

A

Optimal $I(X, Y_N)$

	2	3	4
Newton CG	.8272	1.2925	1.6269
Implicit	.8280	1.2942	1.6291
Greedy	.8280	1.2942	1.6291

B

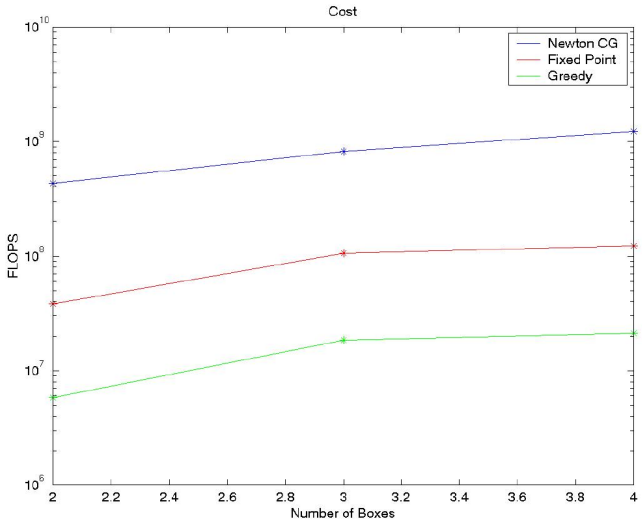

Figure 4: Comparison of the Augmented Lagrangian, implicit solution and vertex search optimization algorithms for $N = 2$, 3, and 4. (A) Compares the value of $I(q(y_N|y)) = I(X, Y_N)$, the mutual information evaluated at the optimal quantizer obtained by each optimization algorithm. (B) A comparison of the computational cost, in FLOPS, incurred by each optimization algorithm for $N = 2$.

4.2 Real Data

4.2.1 Dealing with complex stimuli

To successfully apply our method to physiological data, we need to estimate the information distortion D_I, which in turn depends on the joint stimulus/response probability. If the stimuli are sufficiently simple, $p(x, y)$ can be estimated directly as a joint histogram, and the method is applied as described above. In general, we want to analyze conditions close to the natural for the particular sensory system, which usually entails observing rich stimulus sets of high dimensionality. Characterizing such a relationship non-parametrically is extremely difficult. To cope with this regime, we model the stimulus/response relationship as in [9, 10]. The formulation as an optimization problem suggests certain classes of models which are better suited for this approach. We shall look for models that give us strict upper bounds $\tilde{D}_I$ of the information distortion function D_I. In this case, when we minimize the upper bound $\tilde{D}_I$, the actual value of D_I is also decreased, since $0 \leq D_I \leq \tilde{D}_I$. This also gives us a quantitative measure of the quality of a model: a model with smaller $\tilde{D}_I$ is better.

We start the modeling process by noting that D_I can be expressed as

$$D_I(Y, Y_N; X) = H(X) - H(X|Y) - (H(X) - H(X|Y_N)) \tag{9}$$

by using standard equalities from information theory [5]. The only term in (9) that depends on the quantizer $q(y_N|y)$ is $H(X|Y_N)$, so minimizing D_I is equivalent to minimizing $H(X|Y_N)$. Thus the models we need to consider should produce upper bounds of $H(X|Y_N)$. One way to achieve this is by constructing a maximum entropy model [14] conditioned on constraints

imposed by the researcher. We can express $H(X|Y_N)$ as $H(X|Y_N) = E_{y_N} H(X|y_N)$ [7, 10], where each term $H(X|y_N)$ is the entropy of X conditioned on y_N being the observed response class, and E_{y_N} denotes the expectation in Y_N. As a first attempt, we constrained the class conditional mean $\mu_N(X)$ and covariance $C_N(X)$ of the stimulus to the ones observed from data. The maximum entropy model under such constraints is a Gaussian, $N(\mu_N(X), C_N(X))$. Each entropy term is then bounded by

$$H(X|y_N) \leq H_G(X|y_N) := \frac{1}{2} \log(2\pi e)^{|X|} \det C_N(X)$$

where $|X|$ is the dimensionality of the stimulus space X. This produces an upper bound, $\tilde{H}(X|Y_N)$, of $H(X|Y_N)$:

$$H(X|Y_N) \leq \tilde{H}(X|Y_N) := E_{y_N} H_G(X|y_N) = E_{y_N} \frac{1}{2} \log(2\pi e)^{|X|} \det C_N(X) \tag{10}$$

In this expression, $C_N(X)$ can be expressed explicitly as a function of the quantizer and parameters depending on the data [9, 10]. The stimulus model obtained in this manner is effectively a Gaussian mixture model, with priors $p(y_N)$ and Gaussian parameters $(\mu_N(X), C_N(X))$. We define

$$\tilde{D}_{eff} := \tilde{H}(X|Y_N).$$

Theorem 6 shows that $\tilde{D}_{eff}$ obtained from D_{eff} by this approximation is concave and that the optimal quantizer $q(y_N|y)$ will be deterministic (Corollary 13). This means that $\tilde{D}_{eff}$ can be used in place of D_{eff} in any of the optimization schemes.

4.2.2 Results

A biological system that has been used very successfully to address aspects of neural coding [2, 4, 21, 22, 31] is the cricket's cercal sensory system. It provides the benefits of being simple enough so that all output signals can be recorded, yet sufficiently elaborate to address questions about temporal and collective coding schemes. The cricket's cercal system is sensitive to low frequency, near-field air displacement stimuli [16]. The sense organs are two cerci at the rear of the abdomen. Each cercus is covered by approximately 1000 mechanoreceptor hairs, which are deflected by air currents in the animal's immediate environment. The entire sensory epithelium for this system consists of the 2000 receptors that innervate these hairs. Afferent axons from these receptors project into the terminal abdominal ganglion, where they make synaptic connections to approximately 50 sensory inter-neurons. The entire output layer of this system consists of only 20 of these neurons, which send axons to higher centers.

We apply the method to intra-cellular recordings from identified inter-neurons in the cricket cercal sensory system. During the course of the physiological recording, the system was stimulated with air current stimuli, drawn from a band-limited (5-400Hz) Gaussian white noise (GWN) source [30].

When applying the method to real data, the joint stimulus response probability $p(x, y)$ needs to be estimated. We use (10) $\tilde{D}_{eff}$, an upper bound of the effective distortion D_{eff}, in place of D_I in the optimization scheme [6].

The results in figure 5 show the optimal quantizer for this system. Patterns 2 through 105 in A were obtained by choosing 10 ms sequences from the recording which started with a spike (at time 0 here). Sequences in which the initial spike was preceded by another spike closer than 10ms were excluded. Pattern 2 contains a single spike. Patterns 3-59 are doublets. Patterns 60-105 are triplets. Pattern 1 is a well isolated empty codeword (occurances were chosen to be relatively far from the other patterns). The number of samples from this class were restricted to be comparable to the rest of the set. Each pattern was observed multiple times (histogram not shown).

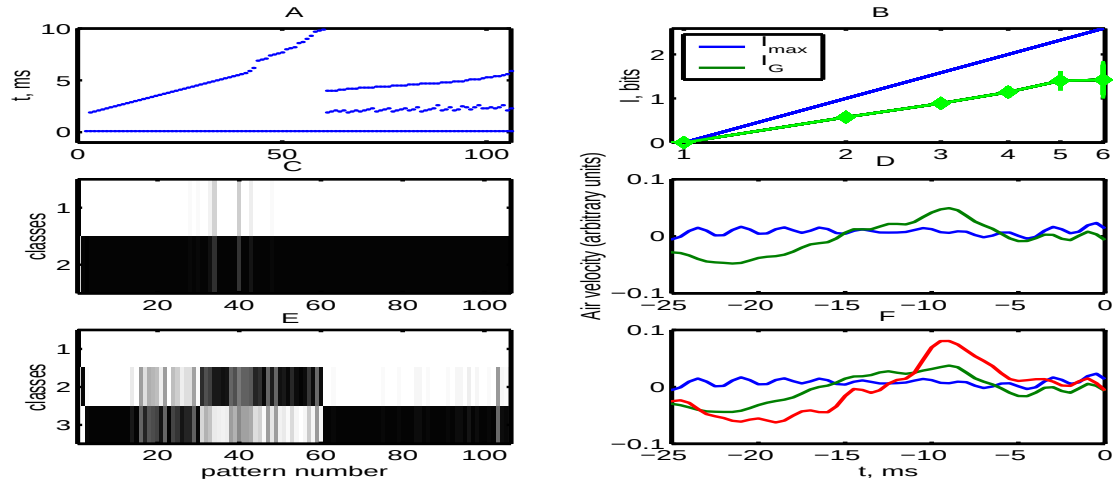

Figure 5: *Results from the information distortion method. (A) Raster plot of Y: All the response spike patterns that were analyzed. Each dot represents the occurrence of a single spike. The bottom row of dots represents the first spike for every pattern. The dots above these represent the subsequent spikes occurring at some time within 10 ms after the first spike. The y axis is the time in ms after the occurrence of the first spike in the pattern. The x axis here and below is an arbitrary number, assigned to each pattern, where all the patterns have been ordered according to number of spikes first, and increasing ISI between spikes second. (B) The lower bound of the I (green) obtained through the Gaussian model can be compared to the absolute upper bound $I = \log_2 N$ for an N class reproduction (blue). (C) The optimal quantizer for $N = 2$ classes. This is the conditional probability $q(y_N|y)$ of a pattern number y, a point on the horizontal axis in A, belonging to class y_N, a point on the vertical axis in A. White represents where $q(y_N|y) = 0$, black represents where $q(y_N|y) = 1$, and intermediate values are represented by levels of gray. (D) The stimulus means, conditioned on the occurrence of class 1 (blue) or 2 (green). (E) The optimal quantizer (obtained by bootstrapping) for $N = 3$ classes. (F) The stimulus means, conditioned on the occurrence of class 1 (blue), 2 (green) or 3 (red).*

Panels C–F show the results of applying the information distortion approach to this dataset. The optimal quantizer for the $N = 2$ reproduction is shown in panel C. It isolates the empty codeword in one class (class $y_N = 1$) and all other patterns in another class (class $y_N = 2$). The mean of the stimuli conditioned with the zero codeword (D, blue), does not significantly deviate from a zero signal.

Panels E and F show the results of extending the analysis to a reproduction of $N = 3$

classes. The zero codeword remains in class 1. The former class 2 is split into two separate classes: class 2, which contains the single spike codeword and codewords with an inter-spike interval $ISI > 5ms$, and class 3, which contains all doublets with $ISI < 2ms$ and all triplets. The mean in (D, green) is split into two separate class conditioned means (F, green and red).

The Augmented Lagrangian algorithm was unable to resolve the optimal quantizer for $N = 3$ and 4. Both the vertex search and the implicit solution algorithms procured optimal quantizers for N = 2,3 and 4 similar to those depicted in Figure 5 C and E. Figure ?? compares the performance of the vertex search algorithm and the implicit solution algorithm on this data set.

5 Theory

The admissible region for the linear constraints in (6) is a direct product of simplices. We show that the optimal solution always occurs at a vertex of this region. This allows us to reformulate (6) as a maximization of the mutual information $I(X, Y_N)$ on the set of vertices. We then describe a new algorithm, which, under mild conditions, always finds a local extremum.

5.1 Maximum on the boundary

In (6), the quantizer $q(y_N|y)$ affects D_I only through $I(X, Y_N)$. Therefore, we pose and investigate the following equivalent maximization problem

$$max_{q(y_N|y)} H(Y_N|Y), \tag{11}$$

with constraints

$$I(X, Y_N) \geq I_0, \tag{12}$$

and

$$\sum_N q(y_N|y) = 1 \quad \text{and} \quad q(y_N|y) \geq 0 \quad \forall y \in Y \tag{13}$$

The parameter I_0 is the informativeness of the quantization. The function is maximized over $q(y_N|y) \in \mathbf{R}^{ns}$, subject to (13), where n is the number of quantization classes and s is the cardinality of the output space Y.

Lemma 1 *The function $I(X, Y_N)$ is a convex function of $q(y_N|y)$.*

Proof. Lemma B.1 of [8]. □

Lemma 2 *Given a convex function $f(x)$, $x \in \mathbf{R}^k$, the set $S(k) := \{x \mid f(x) \leq k\}$ is convex.*

Proof. Let $x_\theta := \theta x_0 + (1-\theta)x_1$. Assume $x_0, x_1 \in S(k)$. Then

$$f(x_\theta) \leq \theta f(x_0) + (1-\theta) f(x_1) \leq k$$

where the first inequality follows from convexity of the function f and the second from the fact that $x_0, x_1 \in S(k)$. □

Let

$$D := \{q(y_N|y) \in \mathbf{R}^{ns} \mid (13) \text{ is satisfied } \}$$

be the set of $q(y_N|y)$ which satisfy the linear constraints. Observe that $D = \Pi_y D_y$ where

$$D_y := \{q(y_N|y) \in \mathbf{R}^{ns} \parallel : \sum_N q(y_N|y) = 1, \quad \text{and } q(y_N|y) \geq 0\}.$$

Each D_y is a standard simplex and D is a product of these simplices.

Let E denote the set of all vertices of the set D. An element e in this set can be written as

$$e = \Pi_y e_y$$

where e_y is a vertex of the simplex D_y.

Lemma 3 *The set D is the convex hull of E.*

Proof. Let $C :=$ convex hull (E). We show first that $D \subset C$. Select a point $w \in D$. Such a point is determined by a collection of barycentric coordinates $s_y^1, \ldots, s_y^n$ in D_y for each y. To show that $w \in C$ we need to find n^s numbers λ_j such that

$$w = \sum_{e(j) \in E} \lambda_j e(j), \quad \sum \lambda_j = 1. \tag{14}$$

We denote the vertices of the simplex D_y by $v_y^1, v_y^2, \ldots, v_y^n$. Observe that (14) will be satisfied if

$$\sum_{e_y(j) = v_y^k} \lambda_j = s_y^k, \quad \text{for all } y = 1, \ldots, s, \quad k = 1, \ldots, n. \tag{15}$$

We construct the set λ_j, satisfying (15), explicitly. We start our construction with a collection of sn barycentric coordinates s_y^k, for $k = 1, \ldots, n$ and for each $y \in Y$, which specify the point w. Let

$$S_y := max_k \; s_y^k$$

for each y and let $m(y) = \text{argmax}_k \; s_y^k$. Hence $S_y = s_y^{m(y)}$. Let $e(1)$ be a vertex of D such that

$$e_y(1) := v_y^{m(y)} \quad \text{for each } y. \tag{16}$$

Finally, we select

$$\lambda_1 := min_y\, S_y.$$

Notice that $\lambda_1 \neq 0$ since for each y at least one $s_y^k \neq 0$. We let $s_y^k(0) := s_y^k$ for all y and all k. We construct a new set of numbers $s_y^k(1)$ (which are no longer barycentric coordinates) in the following way: we replace each number $s_y^{m(y)}$ by number $s_y^{m(y)} - \lambda_1$

$$s_y^{m(y)}(1) := s_y^{m(y)}(0) - \lambda_1. \tag{17}$$

We note two facts about this construction

1. After replacement (17) the sum

$$\sum_{k=1}^{n} s_y^k(1) + \lambda_1 = 1 \quad \text{for each } y.$$

2. At least one number $s_y^{m(y)}(1)$ is zero.

We repeat the construction and in the l-th step we construct vertex $e(l)$, coefficient λ_l, and a new set of numbers $s_y^k(l)$. After the l-th step of the construction we observe that

1. For each fixed y, the sum

$$\sum_{k=1}^{n} s_y^k(l) + \sum_{i=1}^{l} \lambda_i = 1. \tag{18}$$

2.

$$\text{At least } l \text{ numbers } s_y^k(l) \text{ are zero.} \tag{19}$$

Claim 4 *If for $y = t$ there is only one nonzero element s_t^k and for some $y = q$ there are $u > 1$ nonzero elements $s_q^{p_1}, \ldots, s_q^{p_w}$, then in next step of the algorithm s_t^k will not be selected as λ_j.*

Proof. Observe that by (18) above

$$s_t^k = 1 - \sum_{i=1}^{j-1} \lambda_i = \sum_{i=1}^{u} s_q^{p_i}$$

and so the maximum of the set $s_q^{p_i}, i = 1, \ldots, u$ is smaller than s_t^k. □

It follows from the Claim and (19) above that after at most $s(n-1)$ steps the algorithm comes to the situation where for each y there is precisely one $s_y^{k(y)} \neq 0$ and all other s_y^l are zero. Again, by (18), it must be that

$$s_y^{k(y)} = 1 - \sum_{i=1}^{j-1} \lambda_i \text{ for all } y.$$

Hence, in the next step, $\lambda_j := s_t^{k(t)}$ and the algorithm ends. For the vertices $e(i)$ which did not come up in the construction step (16), we set $\lambda_i = 0$. Note that it follows immediately from (18) that

$$\sum_{i=1}^{j} \lambda_i = 1.$$

By construction, $s_y^k(l) \neq s_y^k(l+1)$ for some l if and only if $\lambda_l = s_y^k(l) - s_y^k(l+1)$ and the corresponding vertex $e(l)$ has y-th component, $e_y(l)$, equal to v_y^k. It follows that (15) is satisfied. Hence (14) holds and this proves $D \subset C$.

To show that $C \subset D$ it is enough to realize that D, being a product of convex sets D_y, is convex. Since C is the smallest convex set containing E and $E \subset D$, we have $C \subset D$. □

Theorem 5 *Using the previous notation,*

$$max_E I(X, Y_N) \geq max_D I(X, Y_N).$$

Proof. Denote $M := max_E I(X, Y_N)$ and let

$$A := \{q(y_n|y) \mid I(q(y_n|y)) \leq M\}.$$

By Lemma 1 and Lemma 2, A is a convex set. Since $E \subset A$, then for C, the convex hull of E (and hence the smallest convex set containing E), we have $C \subset A$. By Lemma 3, $C = D$ and thus $D \subset A$. □

5.2 Cost function for real data

In applications to real data, one can either estimate the joint probability $p(x, y)$ and then use the cost function $I(X, Y_N)$, or, as authors in [6] did, one can use a different function, $\tilde{D}_{eff}$, which is an upper bound of $D_{eff} = H(X|Y_N) = E_{y_n} H(X|y_n)$. We note that

$$I(X, Y_N) = H(X) - H(X|Y_N) = H(X) - D_{eff}$$

and the only part which depends on the quantizer $q(y_N|y)$ is D_{eff}. Therefore, maximization of $I(X, Y_N)$ is equivalent to minimization of D_{eff}.

For real data, we estimate the mean and covariance of X conditioned on y_N for each $y_N \in Y_N$. The maximum entropy model for $H(X|y_n)$ under these constraints is Gaussian with estimated mean and covariance matrix $C_{X|y_n}$. Calculation in [6] shows that this leads to an upper bound function $\tilde{D}_{eff}$, (i.e. with the property that $D_{eff} \leq \tilde{D}_{eff}$)

$$\tilde{D}_{eff} = \sum_{y_N} p(y_N) \frac{1}{2} \log(2\pi e)^{|X|} det[\sum_y p(y|y_N)(C_{X|y} + x_y^2) - (\sum_y p(y|y_N) x_y)^2], \qquad (20)$$

where x_y is the mean and $C_{X|y}$ is the covariance matrix of a Gaussian mixture model of X conditioned on a particular y. x_y^2 is a matrix formed by $x_y x_y^T$. Since

$$I(X, Y_N) = H(X) - D_{eff} \geq H(X) - \tilde{D}_{eff} =: \tilde{I}(x, Y_N),$$

this estimate leads to a new version of the optimization problem (11):

$$max_{q(y_N|y)} H(Y_N|Y),$$

with constraints

$$\tilde{I}(X, Y_N) \geq I_0 \tag{21}$$

and (13).

Theorem 6 *The function $\tilde{D}_{eff}$ is concave in $q(y_N|y)$ and hence the function $\tilde{I}$ is convex in $q(y_N|y)$.*

Our argument is based on four Lemmas.

Lemma 7 (Ky-Fan [20]) *The function* log *det A is concave in A.*

Lemma 8 *For all i and j, the (i,j)-th component of the matrix*

$$\mathcal{F} := \sum_y p(y|y_N)(C_{X|y} + x_y^2) - (\sum_y p(y|y_N)x_y)^2$$

is concave in $p(y|y_N)$.

Proof. The first part of $\mathcal{F}$ is linear in $p(y|y_N)$. We look at the second part. Fix i and j and look at the (i,j)-th component of the matrix. After taking out the constants we get that the second part is a function of the form

$$g_{ij} := -(\sum_y a_y p(y|y_N))(\sum_y b_y p(y|y_N)) \tag{22}$$

where a is a vector of i-th components $([x_{y_1}]^i, [x_{y_2}]^i, \ldots, [x_{y_n}]^i)$ and b is a similar vector of j components of x_y. Denote the vector $x := (p(y_1|y_N), p(y_2|y_N), \ldots, p(y_n|y_N))$. Differentiating g_{ij} we arrive at

$$\nabla^2 g_{ij} = -(ba^T + ab^T).$$

The function g_{ij} is concave if the quadratic form

$$x^T(ba^T + ab^T)x$$

is positive semidefinite. Observe that both matrix ba^T and matrix ab^T have rank 1 and so the rank of matrix $M := (ba^T + ab^T)$ is at most two. To show positive semidefiniteness we need to show that the nonzero eigenvalues are nonnegative.

Note that equation $Mv = \lambda v$ leads to

$$(ba^T)v + (ab^T)v = b(a^T v) + a(b^T v) = \lambda v.$$

Since both $(a^T v)$ and $(b^T v)$ are scalars this shows that eigenvectors with nonzero eigenvalues must be in $span\{a, b\}$.

We compute the eigenvalues and eigenvectors by setting $v = c_1 a + c_2 b$ where the constants c_1, c_2 are to be determined.

$$\begin{aligned}(ba^T)v + (ab^T)v &= (ba^T)(c_1 a + c_2 b) + (ab^T)(c_1 a + c_2 b) \\ &= b(c_1(a^T a) + c_2(a^T b)) + a((c_1(b^T a) + c_2(b^T b)) \\ &= \lambda(c_1 a + c_2 b).\end{aligned}$$

Comparing terms in front of a and b (this assumes linear independence of these vectors) we get

$$c_1(a^T a) + c_2(a^T b) = \lambda c_1, c_1(b^T a) + c_2(b^T b) = \lambda c_2.$$

In matrix form, this is $A(c_1, c_2)^T = \lambda(c_1, c_2)^T$, where

$$A = \begin{bmatrix} a^T a & a^T b \\ b^T a & b^T b \end{bmatrix}.$$

So λ is also an eigenvalue of the matrix A. Observe that $a^T a > 0$ and the determinant of A is

$$detA = (a^T a)(b^T b) - (a^T b)^2 \geq 0$$

by Cauchy-Schwartz inequality. So A has nonnegative eigenvalues which are also eigenvalues of M. □.

Lemma 9 *Let $F : \mathbf{R}^n \to \mathbf{R}$ and $f : \mathbf{R}^n \to \mathbf{R}^n$ such that*

1. *$\nabla^2 F$ is negative semidefinite*
2. *If we denote $f = (f_1, f_2, \ldots, f_n)$, then for each i, $x^T \nabla^2 f_i x = 0$.*

Then for $G = F \circ f : \mathbf{R}^n \to \mathbf{R}$ we have that $\nabla^2 G$ is negative semidefinite.

Proof. Straightforward computation shows that

$$\nabla G = Df \; \nabla F(f),$$

where Df is $n \times n$ matrix and both ∇G and ∇F are n vectors. We write out the l-th component of ∇G

$$\frac{\partial G}{\partial x_l} = \sum_{j=1}^{n} \frac{\partial F}{\partial f_j} \frac{\partial f_j}{\partial x_l}.$$

Now compute the (l,k)-th element of the matrix $\nabla^2 G$

$$\begin{aligned}(\nabla^2 G)_{lk} = \frac{\partial}{\partial x_k}(\frac{\partial G}{\partial x_l}) &= \sum_{j=1}^{n}\sum_{s=1}^{n} \frac{\partial^2 F}{\partial f_j \partial f_s} \frac{\partial f_s}{\partial x_k} \frac{\partial f_j}{\partial x_l} + \sum_{j=1}^{n} \frac{\partial F}{\partial f_j} \frac{\partial^2 f_j}{\partial x_k \partial x_l} \\ &= \frac{df}{dx_k} \nabla F \frac{df}{dx_l} + \sum_j \frac{\partial F}{\partial f_j} (\nabla^2 f_j)_{k,l}\end{aligned}$$

Finally, we compute $x^T \nabla^2 G x$:

$$\begin{aligned} x^T \nabla^2 G x &= \sum_l \sum_k (\nabla^2 G)_{l,k} x_l x_k \\ &= \sum_k \sum_l x_k \frac{df}{dx_k} \nabla^2 F \frac{df}{dx_l} x_l + \sum_k \sum_l \sum_j \frac{\partial F}{\partial f_j} (\nabla^2 f_j)_{k,l} x_k x_l \\ &= (Dfx)^T \nabla^2 F (Dfx) + \sum_j \frac{\partial F}{\partial f_j} (x^T \nabla^2 f_j x). \end{aligned}$$

By the second assumption the last term is zero and so

$$x^T \nabla^2 G x = (Dfx)^T \nabla^2 F (Dfx).$$

The first assumption now guarantees that $\nabla^2 G$ is negative semidefinite. □

Lemma 10 *Fix the value of the random variable $y_N = M$. Let*

$$f_i(q(M|y)) := p(y_i|M) = \frac{q(M|y_i)p(y_i)}{p(M)} = \frac{q(M|y_i)p(y_i)}{\sum_j q(M|y_j)p(y_j)}.$$

Then, if we denote $q = (q(M|y_1), q(M|y_2), \ldots, q(M|y_n))$, we have

$$q^T \nabla^2 f_i q = 0$$

for all i.

Proof. To simplify notation we let $a_i := p(y_i)$, $x_i := q(M|y_i)$ and $x = (x_i, \ldots, x_n)$. Then

$$f_i(x) = \frac{a_i x_i}{\sum_j a_j x_j}.$$

We compute

$$\begin{aligned} \frac{\partial f_i}{\partial x_l} &= \delta_{li} \frac{a_l (\sum_j a_j x_j)}{(\sum_j a_j x_j)^2} - \frac{a_l a_i x_i}{(\sum_j a_j x_j)^2} \\ &= \delta_{li} \frac{a_l}{\sum_j a_j x_j} - \frac{a_l a_i x_i}{(\sum_j a_j x_j)^2}, \end{aligned}$$

where $\delta_{li} = 1$ if $l = i$ and zero otherwise. The second derivative is

$$\frac{\partial f_i}{\partial x_l \partial x_k} = -\delta li \frac{a_l a_k}{(\sum_j a_j x_j)^2} - \delta ki \frac{a_l a_k}{(\sum_j a_j x_j)^2} + 2 \frac{a_k a_l a_i x_i x_k x_l}{(\sum_j a_j x_j)^3}.$$

Then $x^T \nabla^2 f_i x$ is

$$\begin{aligned} x^T \nabla^2 f_i x &= \sum_{k,l} \frac{\partial f_i}{\partial x_l \partial x_k} x_k x_l \\ &= \frac{1}{(\sum_j a_j x_j)^2} \left[\sum_k -a_i a_k x_i x_k - \sum_l a_i a_l x_i x_l + \frac{2 a_i x_i}{\sum_j a_j x_j} \sum_{k,l} a_k x_k a_l x_l \right] \\ &= \frac{1}{(\sum_j a_j x_j)^2} (a_i x_i) \left[-\sum_k a_k x_k - \sum_l a_l x_l + \frac{2 \sum_l a_l x_l \sum_k a_k x_k}{\sum_j a_j x_j} \right] \\ &= 0. \end{aligned}$$

Proof of Theorem 6

Lemma 10 and Lemma 8 verify the assumptions of Lemma 9 where we set $f := f_i$ (f_i from Lemma 10) and $F = g_{lk}$ (g_{lk} from Lemma 8), for any k, l, i. Hence, by Lemma 9, each (k,l)-th component g_{lk} of $\mathcal{F}$ is a concave function of $q(M|y_i)$ for all i. By Lemma 7 the function $\log \det \mathcal{F}$ is concave in $\mathcal{F}$ and thus in $q(M|y_i)$ for any i. At this point we should write $\mathcal{F}_M$ instead of $\mathcal{F}$ since we have the value $y_N = M$ fixed in computation of $\mathcal{F}$. Clearly our argument is true so far for any such M. Finally, since $p(y_N) = \sum_y q(y_N|y)p(y)$ is a linear combination of $q(y_N|y)$, then the function $\tilde{D}_{eff} = \tilde{D}_{eff}(q(y_N|y))$ (20 is a linear combination of concave functions

$$\log \det \mathcal{F}_M,$$

where $\mathcal{F}_M$ has fixed value $y_N = M$. This finishes the proof. □

Theorem 11 *Using the previous notation,*

$$max_E \tilde{I}(X, Y_N) \geq max_D \tilde{I}(X, Y_N).$$

Proof. Analogous to the proof of Theorem 5, where we use the Theorem 6 instead of Lemma 1. □

5.3 Equivalent problem

Lemma 12 *Let f be a convex function, $f : D \to R$, where $D := \Delta_1 \times \ldots \times \Delta_k$ and Δ_i is a simplex. Assume that $\max_D f(x) \leq \max_E f(x)$, where E is the vertex set of D, and that $f(e) = k$ for every vertex $e \in E$. Assume also that there exists an interior point p of D such that $f(p) = k$. Then $f(x) = k$ for all $x \in D$.*

Proof. Fix a set $U := \mathrm{Int}(\Delta_1 \times \ldots \times \Delta_k)$, $U \subset D$. Let $A := \{x \in U \mid f(x) = k\}$. This set is clearly closed since $A = f^{-1}(k)$ and f is continuous. We show that A is open in U. Let us first consider $x \in A$. Since U is open, there is an open neighborhood $N(x) \subset U$. Pick an arbitrary $y \in N(x)$. Since $N(x)$ is open there is a $z \in N(x)$ such that

$$(y + z)/2 = x.$$

By convexity $k = f(x) \leq f(y)/2 + f(z)/2$. By assumption $f(y) \leq f(x)$ and $f(z) \leq f(x)$ and so

$$f(x) \leq f(y)/2 + f(z)/2 = f(x) = k.$$

It follows that $f(x) = f(y) = f(z) = k$. Hence if $x \in A$ then $N(x) \subset A$. Since every U is connected, either $A = U$ or $A = \emptyset$. By assumption $p \in A$ and so $A = U$. By continuity of the function f

$$f(x) = k \quad \text{for all} \quad x \in D.$$

□.

Corollary 13 *The optimal solution of the problem (11) with constraints (12) and (13) can be found by the following algorithm:*

1. *Find a vertex $e \in E$ such that*

$$I(e) := \max_E I(X, Y_N)$$

2. *if all neighboring vertices e_i (which differ from e in exactly one entry, e_y) have $I(e_i) < I(e)$, then e is an optimal solution of (11) with (12) and (13).*

3. *if there is a set of vertices $e_1, \ldots, e_k$ such that $I(e_i) = I(e)$, consider the region $D := \Delta_1 \times \ldots \times \Delta_k$ spanned by these vertices. Pick a point $x \in \mathrm{Int}D$. If $I(x) = I(e)$ then the solution of (11) is the product of the barycenters of Δ_i. If $I(x) < I(e)$ then any vertex e_i is a solution of (11).*

Moreover, the problem for real data (11) with constrains (21) and (13) is equivalent to the problem described above, where the function I is replaced by function $\tilde{I}$.

Proof. **Add proof for (1) and (2)** (for (3)) If $I(x) = I(e)$ then by Lemma 12 $I(y) = I(e)$ for all $y \in D$. Then the solution with maximum entropy is the product of barycenters of Δ_i. If $I(x) \neq I(e)$ by Theorem 5 $I(x) < I(e)$ and by Lemma 12 $I(y) < I(e)$ for all $y \in D$. Result follows.

The only difference in the proof for the function $\tilde{I}$ is that we use Theorem 11 instead of Theorem 5. □

5.4 Vertex Search Algorithm and convergence to a local maximum

All results in this section are valid for both I and $\tilde{I}$. We will mention only I in the the text.

Vertex Search Algorithm (23)

Description:

1. Start at any initial point in D. One possible choice is to start at the point $q(y_N|y) = 1/N$ for all y as in Figure 6A.

2. Select randomly y_1 and evaluate mutual information at all the vertices of D_{y_1}, so that $q(L|y_1) = 1$ for some class $y_N = L$ and zero for all other classes M. Select the assignment of y_1 to a class which gives the maximal mutual information. See Figure 6B.

3. repeat step 2 with $y_2, y_3, \ldots$ until all y_k are assigned classes. This yields a vertex e of D. See Figure 6C.

Remark 14 Clearly the assignment of y_1 to a class is arbitrary, so the algorithm should start with y_2 after y_1 is assigned to a class randomly.

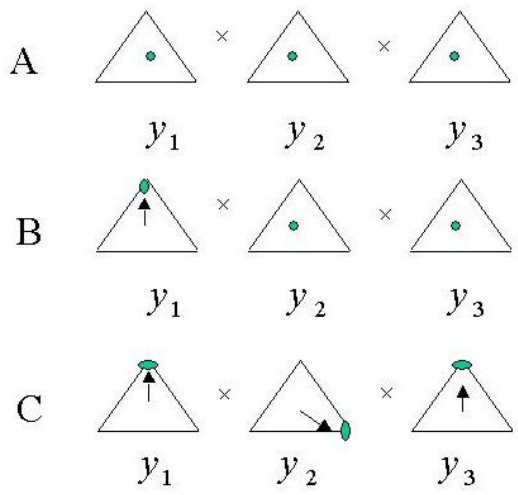

Figure 6: **The vertex search algorithm**, shown here for $N = 3$ and $|Y| = s = 3$. (A) The algorithm begins at some initial $q(y_N|y)$. Here, we start with $q(y_N|y) = 1/3$ for all y and y_N. (B) Randomly assign y_1 to each class: $y_N = 1$, 2 and 3. For each of these classifications, evaluate $I(X, Y_N)$. Assign y_1 to the class y_N which maximizes the mutual information. (C) We repeat the process in (B) for y_2 and then for y_3. Shown here is a possible classification of y_1, y_2 and y_3: y_1 and y_3 are put into one class (call it $y_N = 1$), and y_2 is put into another class (call this one $y_N = 2$). $y_N = 3$ remains empty.

Theorem 15 *The point e is a local maximum of I if for each k and each class L, the numbers*

$$p(x, y_k) << \sum_{y_i \in L, i \neq k} p(x, y_i), \quad p(y_k) << \sum_{y_i \in L, i \neq k} p(y_i).$$

Proof. Assume that the points y_i, $i = 1, \ldots, k-1$ were assigned to their prospective classes by steps 2 and 3. The algorithm decides where to assign y_k based on the mutual information of different assignments at this point. We can write $I(y_k \to L)$ for the value of the mutual information when we assign $q(L|y_k) = 1$ and $q(M|y_k) = 0$ for $M \neq L$.

Let

$$S(L, x) := \frac{\sum_y q(L|y)p(x, y)}{p(x) \sum_y p(y)}.$$

We denote $S_L(N, x)$ the function $S(N, x)$ where we assigned y_k to class L (i.e. $q(y_N =$

$L|y_k) = 1$ and zero otherwise). We compute

$$\begin{aligned} I(y_k \to L) &= \sum_{x,y,y_N} q(y_N|y)p(x,y)\log S(y_N,x) \\ &= \sum_{y_N}\sum_{x}\log S(y_N,x)(\sum_{y_i\in L} p(x,y)) \\ &= \sum_{y_N\neq L}\sum_{x}\log S(y_N,x)(\sum_{y_i\in L, i\neq k} p(x,y)) \\ &+ \sum_{x}\log S_L(L,x)(\sum_{y_i\in L, i\neq k} p(x,y)+p(x,y_k)) \end{aligned} \tag{24}$$

We select $q(L|y_k) = 1$ if and only if

$$d_{LM} := I(y_k \to L) - I(y_k \to M) \geq 0$$

for all $M \neq L$. Observe that the first term in (24) is the same for $I(y_k \to L)$ and $I(y_k \to M)$. Then d_{LM} for fixed classes L and M is

$$\begin{aligned} d_{LM} &= \sum_{x}\log S_L(L,x)[\sum_{y_i\in L, i\neq k} p(x,y)+p(x,y_k)] \\ &+ \sum_{x}\log S_L(M,x)[\sum_{y_i\in M, i\neq k} p(x,y)] \\ &- \sum_{x}\log S_M(L,x)[\sum_{y_i\in L, i\neq k} p(x,y)] \\ &- \sum_{x}\log S_M(M,x)[\sum_{y_i\in M, i\neq k} p(x,y)+p(x,y_k)] \end{aligned} \tag{25}$$

Denote $\epsilon_1 := \frac{p(x,y_k)}{\sum_{y\in M} p(x,y)}$ and $\epsilon_2 := \frac{p(y_k)}{\sum_{y\in M} p(y)}$. We compute

$$\begin{aligned} \frac{S_L(M,x)}{S_M(M,x)} &= \frac{\sum_{y\in M} p(x,y) + 1/N\sum_{i<k} p(x,y_i)}{\sum_{y\in M, y\neq k} p(x,y)+p(x,y_k)} \frac{\sum_{y\in M, y\neq k} p(y)+p(y_k)}{\sum_{y\in M, y\neq k} p(y)} \\ &= (\frac{1}{1+\epsilon_1})(1+\epsilon_2) \\ &\approx 1-\epsilon_1\epsilon_2. \end{aligned}$$

Similarly,

$$\frac{S_L(L,x)}{S_M(L,x)} \approx 1.$$

Then from (25) we get

$$\begin{aligned} d_{LM} &= \sum_x p(x,y_k)[\log S_L(L,x) - \log S_L(M,x)] \\ &+ \sum_x \log \frac{S_L(L,x)}{S_M(L,x)}[\sum_{y\in L, y\neq k} p(x,y)] \\ &+ \sum_x \log \frac{S_L(M,x)}{S_M(M,x)}[\sum_{y\in M, y\neq k} p(x,y))] \\ &= \sum_x p(x,y_k)[\log S_L(L,x) - \log S_L(M,x)] + O(\epsilon_1\epsilon_2). \end{aligned} \tag{26}$$

Now we look at conditions under which a vertex e is a local maximum of the function $I(q(y_N|y))$. These conditions are equivalent to the Karush-Kuhn-Tucker conditions for a local maximum. At a point where $I(q(y_N|y))$ achieves a local maximum the projection of the gradient ∇I onto each affine space forming the boundary of D must fall outside D. A boundary near a vertex is a collection of affine faces, each spanned by the vectors $e - e_i$, where e_i is a vertex of D which differs from e in i-th component only. If the projection

$$(\nabla I)_e \cdot (e - e_i) \geq 0 \tag{27}$$

for all i then e is a local maximum. For each i there are N vectors $e_i = \{e_i^L\}_{y_N = L}$. From [8],

$$(\nabla I)_{q(L|\bar{y})} = \sum_x p(x,\bar{y}) \log S(L,x).$$

Select $y = y_i$. Assume that at the vertex e we have $q(L|y_i) = 1$. Taking the dot product of ∇I with the vector $e - e_i^M$, where the gradient is evaluated at the point e gives

$$(\nabla I)_e \cdot (e - e_i^M) = \sum_x p(x,y_i)[\log S_L(L,x) - S_L(M,x)]. \tag{28}$$

Observe that in the course of the algorithm $q(L|y_k)$ is selected to be 1, if and only if $d_{LM} \geq 0$ for all $M \neq L$. This condition, by (26), is equivalent to condition for local maximum ((27) with (28)) at the point e. □

To satisfy the conditions of Theorem 15 we suggest the following improvement of the vertex search algorithm:

1. Assign y_k to classes L in such a way that

$$\sum_{y_k \in L_j} p(y_k) \approx 1/N$$

where N is the number of desired classes. This choice represents a vertex in D. This choice can be made by ordering all probabilities $p(y_j)$ according to size and then assigning them to classes in order $1, 2, 3, \ldots, N, N, N-1, \ldots, 2, 1, 1, \ldots$.

2. Select randomly y_1 and evaluate the mutual information at all the vertices of D_{y_1}, so that $q(L|y_1) = 1$ for some class $y_N = L$ and zero for all other classes M. Select the assignment of y_1 to a class which gives the maximal mutual information.

3. repeat step 2 with $y_2, y_3, \ldots$ until all y_k are assigned classes. This yields a vertex e of D.

6 Conclusions

The intersting thing to note ...

References

[1] H. B. Barlow. Possible princilples underlying the transformation of sensory messages. In W. A. Rosenblith, editor, *Sensory Communications*. MIT Press, Cambridge, MA, 1961.

[2] D. A. Bodnar, J. Miller, and G. A. Jacobs. Anatomy and physiology of identified wind-sensitive local interneurons in the cricket cercal sensory system. *J. Comp. Physiol. A*, 168:553–564, 1991.

[3] L. Breiman. *Probability*. Addison-Wesley Publishing Companyy, Menlo Park, CA, 1968.

[4] H. Clague, F. Theunissen, and J. P. Miller. The effects of adaptation on neural coding by primary sensor interneurons in the cricket cercal system. *J. Neurophysiol.*, 77:207–220, 1997.

[5] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley Series in Communication, New York, 1991.

[6] A. G. Dimitrov and J. P. Miller. Analyzing sensory systems with the information distortion function. In R. B. Altman, editor, *Pacific Symposium on Biocomputing 2001*. World Scientific Publushing Co., 2000.

[7] A. G. Dimitrov and J. P. Miller. Natural time scales for neural encoding. *Neurocomputing*, 32-33:1027–1034, 2000.

[8] A. G. Dimitrov and J. P. Miller. Neural coding and decoding: communication channels and quantization. *Network: Computation in Neural Systems*, 12(4):441–472, 2001.

[9] A. G. Dimitrov, J. P. Miller, and Z. Aldworth. Neural coding and decoding. New Orleans, November 2000. Society for Neuroscience Annual Meeting.

[10] A. G. Dimitrov, J. P. Miller, Z. Aldworth, and A. Parker. Spike pattern-based coding schemes in the cricket cercal sensory system. *Neurocomputing*, 2002. *(to appear)*.

[11] A. Gersho and R. M. Gray. *Vector Quantization and Signal Compression*. Kluwer Academic Publishers, 1992.

[12] R. M. Gray. *Entropy and Information Theory*. Springer-Verlag, 1990.

[13] D. Hubel and T. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol. (London)*, 195:215–243, 1961.

[14] E. T. Jaynes. On the rationale of maximum-entropy methods. *Proc. IEEE*, 70:939–952, 1982.

[15] D. H. Johnson, C. M. Gruner, K. Baggerly, and C. Seshagiri. Information-theoretic analysis of the neural code. *J. Comp. Neurosci*, 10(1):47–70, 2001.

[16] G. Kamper and H.-U. Kleindienst. Oscillation of cricket sensory hairs in a low frequency sound field. *J. Comp. Physiol. A.*, 167:193–200, 1990.

[17] C. T. Kelley. *Iterative Methods for Optimization*. SIAM, Philadelphia, 1999.

[18] T. W. Kjaer, J. A. Hertz, and B. J. Richmond. Decoding cortical neuronal signals: Network models, information estimation and spatial tuning. *J. Comp. Neurosci*, 1(1-2):109–139, 1994.

[19] S. Kullback. *Information Theory and Statistics*. J Wiley and Sons, New York, 1959.

[20] Ky-Fan. On a theorem of weyl concerning the eigenvalues of linear transformations ii. Proc. NAtional. Acad. Sci. U.S., 36:31–35, 1950.

[21] M. A. Landolfa and J. P. Miller. Stimulus-response properties of cricket cercal filiform hair receptors. *J. Com. Physiol. A.*, 177:749–757, 1995.

[22] J. P. Miller, G. A. Jacobs, and F. E. Theunissen. Representation of sensory information in the cricket cercal sensory system. I. Response properties of the primary interneurons. *J. Neurophys*, 66:1680–1689, 1991.

[23] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, 2000.

[24] P. Reinagel and R. Reid. Temporal coding of visual information in the thalamus. *J. Neurosci.*, 20(14):5392–5400, 2000.

[25] F. Rieke, D. Warland, R. R. de Ruyter van Steveninck, and W. Bialek. *Spikes: Exploring the neural code*. The MIT Press, 1997.

[26] K. Rose. Deteministic annealing for clustering, compression, classification, regerssion, and related optimization problems. *Proc. IEEE*, 86(11):2210–2239, 1998.

[27] C. E. Shannon. A mathematical theory of communication. *Bell Sys. Tech. J.*, 27:623–656, 1948.

[28] S. P. Strong, R. Koberle, R. R. de Ruyter van Steveninck, and W. Bialek. Entropy and information in neural spike trains. *Phys. Rev. Let.*, 80(1):197–200, 1998.

[29] F. Theunissen and J. P. Miller. Temporal encoding in nervous systems: A rigorous definition. *J. Comp. Neurosci*, 2:149–162, 1995.

[30] F. Theunissen, J. C. Roddey, S. Stufflebeam, H. Clague, and J. P. Miller. Information theoretic analysis of dynamical encoding by four primary sensory interneurons in the cricket cercal system. *J. Neurophy.*, 75:1345–1359, 1996.

[31] F. E. Theunissen and J. P. Miller. Representation of sensory information in the cricket cercal sensory system. II. Information theoretic calculation of system accuracy and optimal tuning curve width of four primary interneurons. *J. Neurophysiol.*, 66:1690–1703, 1991.