ERC Report - May 17, 2010

Amir Hesam Salavati
E-mail hesam.salavati@epfl.ch

Supervisor: Prof. Amin Shokrollahi
E-mail amin.shokrollahi@epfl.ch
Algorithmics Laboratory (ALGO)
Ecole Polytechnique Federale de Lausanne (EPFL)

May 17, 2010

1 Introduction

In this report, I will give a summary of papers I have read during past week. First, a paper on biological mechanisms of error detection is reviewed. Afterward, I will discuss some papers on error correction codes in neural networks. Information-theoretical approaches in investigating neural code properties is another subject which will be briefly discussed. Finally, I review some of my ideas for further works on applications of coding theory in neural systems.

2 Biological Mechanisms Involved in Error Detection

In [1], neuroimaging techniques (fMRI in particular) was used to identify parts of brain that are involved in error detection in behavior. Here, error detection refers to the ability of brain to monitor and ongoing activity and detect an error if the outcome does not match the correct expected result. Therefore, it is quite different from what engineers mean by error detection in communication systems. However, the fact that error detection could happen even if the person doing the task is unaware of it, makes the results in [1] more interesting.

To assess the error detection capability of one's brain, an experiment was conducted upon thirteen persons. A series of consecutive color names were shown to each subject where in some cases, the color of the word and the its did not match. In these cases, the subject should push a button declaring an error. Meanwhile, some electrode were also recording signals from his/her brain activity.

Results show that in 70% of instances, subjects were aware of errors. Moreover, fMRI scans identifies parts of one's brain involved in error detection. Based on the results, Prefrontal (PFC) and anterior cingulate (ACC) cortices are critical to error processing. In addition, the results show that ACC may know more about our ongoing performance than we ourselves are aware of [1].

3 Error Correction and Neural Coding

The main job of sensory neurons is to encode stimuli efficiently in a way that its neighbor neurons be able to decode the resulting spike train and get an estimate of the stimulus as close to the original one as possible. However, the channel between two neurons is prone to noise (because of axon and synapses). Thus, it seems reasonable that neurons employ error correction techniques in encoding stimulus. This becomes even more important for a group of neurons and population codes.

In [2], authors have considered this problem and investigated cases in which a group of neurons use block coding techniques to enhance error correction abilities of the neural code. To be more specific, they have considered a system in which data is encoded linearly via an encoding matrix, W. Encoded data is transmitted over a noisy channel with additive noise. The noisy sequence is then decoded using a decoding matrix, A. This process gives us an estimate of the input data, $\widehat{x}$. Systems model is shown in figure 1.

The goal is to construct A such that the reconstruction error, $E[(\widehat{x}-x)^2]$ is minimized subject the power constraint (or equivalently, SNR constraint) in an AWGN channel. The authors have analytically solved this problem to find the optimal decoding matrix for $N = 1$ and $N = 2$, where N is the dimension of x. For larger N, a numerical approach is used.

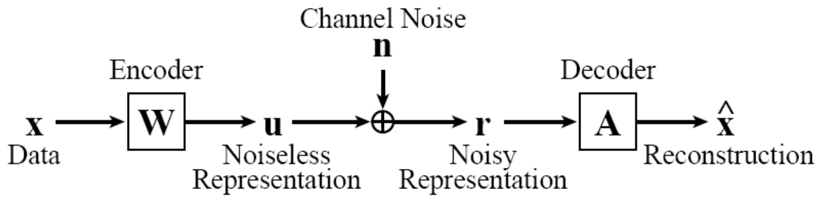

Figure 1: The model used in [2]

The proposed method was also employed to design new image encoding mechanisms the power of which was tested via simulations to show its superiority over current image encoding methods such as ICA and Wavelet. However, a major weakness of the paper is the fact that the suggested method was not applied to biological systems, specially brain, and was not compared to biological experimental results. Such comparisons could be very important in investigating error correction properties of brain.

Another promising application of coding theory is in designing artificial neural networks to overcome noise and be able to correct errors as much as possible. One such application is in designing associative memories [3]. In the last report, I mentioned that an associative memory is a part of brain responsible for pattern classification and correctly recalling impaired patterns. Therefore, it performs some sort of error correction to correctly match noisy patterns to a previously memorized one. Artificial associative memories have the same task. In [3], authors determine required conditions on weight matrix W of a Bidirectional Associative Memory (BAM) such that it is able to correct any errors in the input sequence if the Hamming weight of the error is less than a maximum determined by the training pairs.

BAM is a two-layer hetero-associative feedback neural network model. In a BAM we have an input layer L_A with n neurons and an output layer L_B with m neurons. We have both forward and backward connections between these two layers but no interconnection inside a layer. More over, input and output of BAM are binary. Figure 2 illustrates the model.

In original BAM, data is memorized according to the following equation

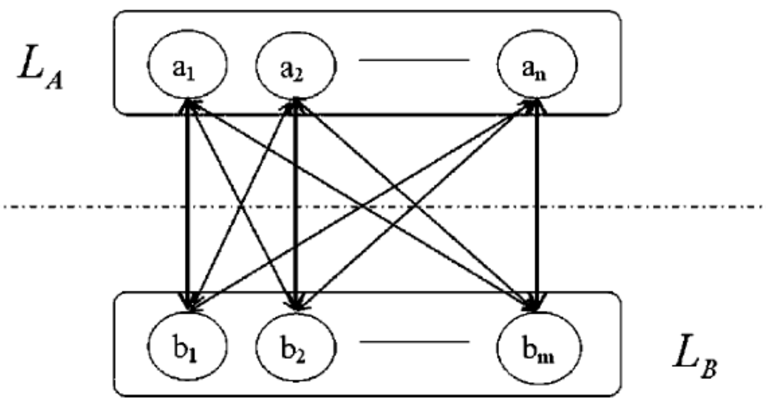

Figure 2: Schematic model of a BAM [3]

which gives us the correlation matrix averaged over all training pairs:

$$M = \sum_{i=1}^{N} X_i^T Y_i \tag{1}$$

Where N is the number of training pairs and X_i and Y_i are bipolar representation of A_i and B_i, i.e. we have $+1/-1$ in X_i instead of $0/1$ as in A_i. Now if training pairs are orthogonal ($X_i X_j = 0$ for $i \neq j$), then we clearly see that feeding the network with input pattern X_i, we get the corresponding output pattern,Y_i: $X_i M = Y_i$. However, if input is noisy, then the corresponding correct output is not necessarily returned by the above process.

To overcome the effect of noise, we have to use weighted BAM in which correlation matrix is the weight sum of input patterns:

$$M = \sum_{i=1}^{N} w_i X_i^T Y_i \tag{2}$$

Now our goal is to find W such that BAM is able to tolerate largest amount of noise, i.e. being able to correct errors with Hamming distances as large as possible. This goal is accomplished in theorem 1 of [3] in which optimal

W is find using genetic algorithms. The argument used to derive optimal W is very similar to coding-theoretical arguments to assess error correction capabilities of a code.

The result were also experimentally tested on an artificial BAM to distinguish between three patterns. All training pairs have been recalled correctly and all noisy input pairs with less than 4 Hamming distance away from the training pairs have converged to the correct training pair. We also find a pattern with 5 Hamming distance away from the training pair 1, which cannot be recalled correctly. Moreover, the proposed method was also compared to other approaches and simulation results show that the suggested method can find the maximum noise tolerance set, which is not guaranteed in other algorithms [3]. Extension of this and similar approaches to biological neuronal networks seems to be a promising area.

4 Information Theoretical Approaches in Analyzing Neural Coding

Information theory has been used in analyzing neural codes for a long time. A recent and interesting example of such applications is mentioned in [4]. The goal of this paper is to study neural coding from a new perspective. To do so, neural sensory system is modeled as a communication channel. Based on this model, authors have used jointly **typical sequences**' theory to investigate properties of neural codes. In this viewpoint, neural code has a structure like a dictionary: there are sets of stimulus-response pairs that are synonyms. Moreover, these sets are independent except for a few common members between each set. In order to identify this dictionary, quantization is used to reduce the size of data set. Among all possible quantization, the one which preserves as much information as possible among stimulus-response pairs are used.

The model considered in [4] is similar to models in other problems in information theory: we have an input with probability distribution $p(x)$. This input is transmitted over a channel with probability distribution of $p(y|x)$ which results in the output y with probability distribution of $p(y)$. Here, input is the stimulus, channel is the neuron and output is the spike train. From typical coding theory, we expect that the received sequence, y^n, and the transmitted sequence, x^n make a joint typical set and we can use typical coding/decoding theorems to analyze this problem. Those input and output sequences that are jointly typical form an equivalency class. In this regard, many neural codes used in neuronal systems are a special case of this model. For example in rate coding, all sequences that have the same number of spikes in a given interval are members of the same equivalency class.

To identify the equivalence classes, we have to find the mutual information, I, and conditional entropy, H, between x^n and y^n. A big challenge in estimating H and I is the large size of bulk of data needed to closely approximate the probability distribution. The number of data points for good approximations actually grows exponentially with correlation period T and number of neurons K [4]. To solve this issue, the quantized version of data is considered. Among all possible quantization schemes, the one which preserves as much information between stimulus and response as possible is selected. A stochastic quantizer is used to map the output sequence Y to the quantized version Y_N with N quantization levels according to the (to be determined) distribution $q(y_N|y)$. The quality of quantization is assessed with the following measure [4]:

$$D_I(Y;Y_N) = I(X;Y) - I(X;Y_N) \tag{3}$$

Based on this model, rate coding is simply equivalent to a deterministic quantizer. Now the problem is min DI which is equivalent to the following problem [4]:

$$\max_{q(Y_N|Y)} H(Y_N|Y) \tag{4}$$

subject to

$$D_{eff} = I(q(Y_N|Y)) \geqslant I_0 \tag{5}$$

and

$$\sum_{Y_N} q(Y_N|Y) = 1 \tag{6}$$

Where I_0 is a threshold determining the precision of the quantizer.

This is a concave problem so the solution lies at the boundaries, i.e. $I(q(y_N|y)) = I_0$. Hence, by pushing I_0 further and further, we reach a point, I_0^{max}, beyond which the above problem does not have a solution. This is the maximum possible information we can get for a fixed number of quantization levels, N.

Authors have used multiple approaches to solve the above problem and to approximate it in different ways. We know that $I(x;y_N)$ ¡= $I(x;y)$ since

x, y and y_N make a markov chain. We would like to maximize $I(x; y_N)$ to minimize D_I. We also know that $I(x; y_N) \leqslant \log_2 N$. Therefore, we would like to increase N so to maximize $I(x; y_N)$. However, if we keep increasing N, we reach a point, N_c, where becomes larger than $I(x; y)$, i.e. $N_c > I(x; y)$. Hence, increasing N beyond this value does not contribute to $I(x; y_N)$.

Now here is how we deduce the codebook: we fix an appropriate and acceptable amount of distortion, D_I. This gives us N. Having N, we start building $q(y_j|y)$, which is the probability of a particular y being a member of j^{th} equivalency class (= quantization level). Having that, we can build a similar distribution $q(y_j|x)$ which is the same probability for input sequences. If D_I is small, then the responses associated with class y_N are

$$\mathrm{y_N} = \{y|q(y_N|y) \simeq 1\} \tag{7}$$

In this case, if we group all those x's whose output belongs to the same equivalency class, i.e.:

$$\mathrm{x_N} = \{x|q(y_N|x) \simeq 1\} \tag{8}$$

We end up having equivalency classes in x's. Therefore, we will have probability distribution $q(y_i|x_j)$, where x_j is the j^{th} equivalency class. This gives us the codebook for the quantized model.

The above method was applied to syntectic data, which is one of the main drawbacks of this paper. Nevertheless, the results of experiments on artificial data are promising.

The method is also applied to the case where x and y are input and output of a Hamming(7,4) code. Then, noise is applied to both x and y to get x_0 and y_0. This is just to make a non-deterministic relationship between x and y which gives us something like equivalency classes as we already discussed. Therefore, this has nothing to do with evaluating the error correction properties of code. Instead, the goal is to find the coding structure of the neural code (the mapping between stimulus and response). The results illustrated in figure 3 show that the proposed approach is able to identify jointly typical classes which will be extremely helpful in analyzing neural codes.

Applying this method to actual neural data seems a natural step in investigating coding properties of neuron. Because if error correction codes are used by neurons, then equivalency classes must exist in the output spike train of neurons. These classes correspond to spheres around valid codewords. Therefore, if application of this method to neural data results in

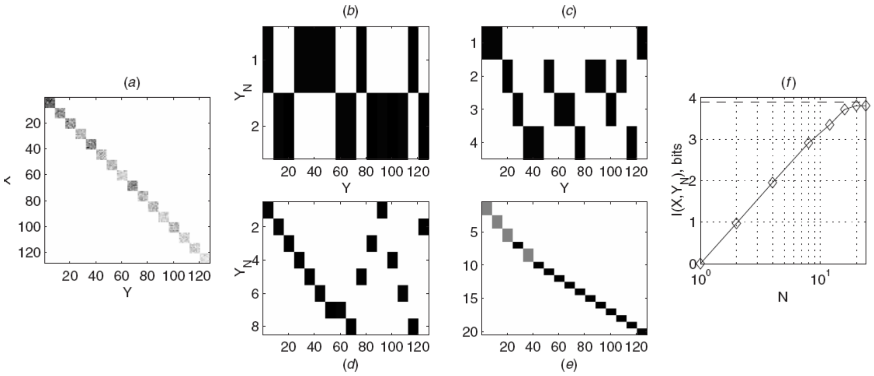

Figure 3: The joint probability between X and Y after permuting rows and columns (a) and optimal quantization for different number of classes (b)(e). The behavior of the mutual information with increasing N can be seen in the loglinear plot (f). The dashed curve is $I(X;Y)$ [4].

pattern similar to those presented in figure 3, it could be a sign of error correction codes in neural code.

Another application of information theory in analyzing population neural code is mentioned [5]. Although information theory has only a minor role in this work, the approach used here is very promising and clearly shows the effect of population coding in reducing the reconstruction error (MES) between the estimated stimulus and the actual one. However, this work mostly concerns source coding and how good a source coder neurons are.

There are, at least theoretically, two types of behavior when it comes to collective encoding of a stimulus in neuronal networks: independent neurons and coupled neurons. In the first case, neurons encode the stimulus independently. However, any correlation in the stimulus causes the output of these neurons to be highly correlated. In the later case, the firing pattern of a neuron not only depends on the stimulus (and its history), but also on the firing patterns of other neurons. Therefore, the outputs of neurons are correlated, both because of correlations in stimulus and coupling among neurons. Whether the coupling among neurons (which is called functional connectivity here) are a result of population coding is investigated in this paper from an information theoretical point of view. The functional connectivity could be viewed as a sort of cooperation among neurons to collectively encode stimulus.

In a nutshell, here is how authors assess efficiency of neural code from a source coding point of view in [5]: first, a fully connected network is considered. A training task is performed over this network such that some connections are strengthened and some other are weakened according to a variation of Hebbian plasticity rule (see previous report). Then, the resulting network structure for a given task (stimulus) is infered by determining the strength of the connection between each neuron. To do so, the following parameter is used:

$$\Delta I_k(i,j) = I(r_i; r_j | s_k) - I(r_i; I_j) \tag{9}$$

Which is the difference between the average mutual information between neurons i, j and their MI when stimulus s_k is applied. The larger this parameter is, the stronger the connection between neurons i and j would be for stimulus s_k. Finally, to evaluate the performance of the system, we try to rebuild the stimulus according to the network model and then compare it with the actual stimulus:

$$\widehat{s}_k = argmax_k \{P(s_k|\mathbf{r})\} \tag{10}$$

where $\mathbf{r}$ is the rate vector of all neurons and $P(s_k|\mathbf{r})$ is the conditional probability of having had stimulus s_k if the rate vector is equal to $\mathbf{r}$. The lower the MSE between s and $\widehat{s}$is, the better the system would be. Note that a different kind of network may be observed for different tasks (to get a better estimate of the stimulus). F

What the authors have done is to simply evaluate this model and compare it with other models to show that it is superior to all of them according to MSE criterion. Results are shown in figure 4 clearly indicate the superiority of this model compared to other approaches.

Optimal population (source) coding was also addressed in [6]. The goal of this paper is to answer the following question: what is an optimal population coding? To be more specific, the goal is to find out how firing rate and dependencies among a group of neurons should change as the stimulus varies over its entire range such that the estimated stimulus from these firing rates be as close to the actual stimulus as possible. optimal (source) coding is achieved when the output of neurons are independent. However, due to correlations in the stimulus, this not accomplished unless neurons cooperate. This is one reason for why population coding is important.

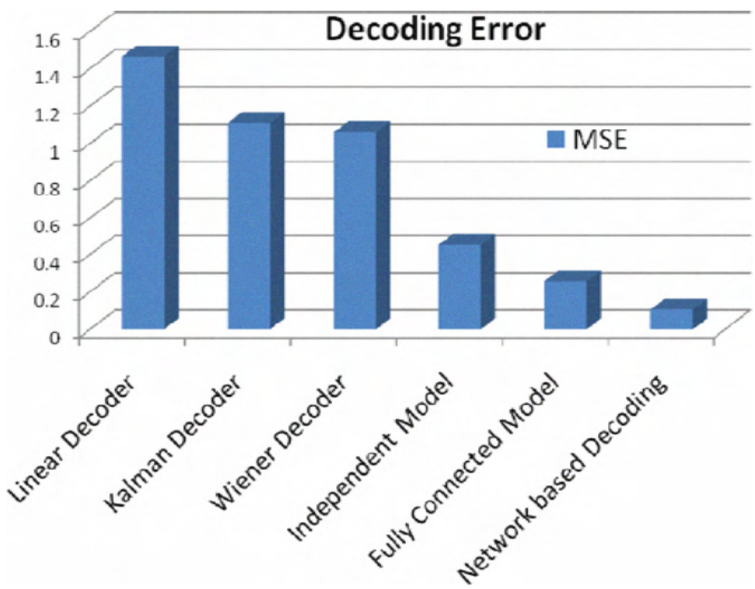

Figure 4: Comparison between decoding error of different approaches [5].

Here, we seek the optimal cooperation strategy. That's the strategy that leads to least MSE, which is the error between the estimated stimulus attribute (for example the velocity of a moving object) and the actual attribute. Moreover, the stimulus attribute is assumed to be a scalar. The optimal cooperation strategy gives us $p(y|x)$ which results in out estimation of stimulus. So the main task is to determine the optimal $p(y|x)$.

To tackle this problem, Fisher information was used. Fisher information is important because of the Cramer-Rao bound, which states that the mean squared error is lower bound by the reciprocal of Fisher information. Various models were considered in [6] and results show that population coding can perform as a better source coder compared to cases where neurons act independently. To be more particular, if each neuron in a group of neurons receives input that is statically independent of the others, then non-cooperative neurons achieve optimal coding. However, when there is correlation among the inputs of different neurons, which happens whenever there is correlation in the stimulus, then non-cooperative population coding is not optimal because we will have stimulus-induced correlation in the output. In such cases, cooperative population codes that decorrelate input and give statistically independent outputs are optimal. Therefore, we need cooperation among neurons to eliminate correlations in the stimulus. This could accomplished by adding a second stage of cooperative neurons after a stage of non-cooperative population coding. Analyzing population codes is one of the the areas in neuroscience in which information theory has been used with success [7].

Although information theory is used quite extensively in neuroscience, some researchers argue that classical information theory is not suitable for neuroscientific problems since information theory was developed to design appropriate communication systems. However, in neuroscience, we would like to analyze pre-designed system [7]. Nevertheless, most of the comments in this paper seems irrelevant when one considers information theoretical approaches used in analyzing neural systems. A simple example is the case where information theory is used to assess the efficiency of neural codes by calculating the channel capacity of a neuron and compare it with the actual rate. Another example is successful information-theoretical methods such as the one mentioned in [4]. Nevertheless, author of [7] believes that information theory could be even more useful if it extended to use Kullback-Leibler distance" instead of mutual information in analysis (similar to the approaches used in [4] and [6]). Moreover, information theory is used quite successfully in analyzing population neural codes [7].

5 Ideas for Future Work

Based on the works reviewed in this report, the following subjects seem to be promising in investigating coding properties of neural systems:

- As mentioned before, applying the method introduced in [4] to actual neural data seems a natural step in investigating coding properties of neuron. Because if error correction codes are used by neurons, then equivalency classes must exist in the output spike train of neurons. These classes correspond to spheres around valid codewords. Therefore, if application of this method to neural data results in pattern similar to those presented in figure 3, it could be a sign of error correction codes in neural code.

- In designing channel codes, we artificially introduce redundancy to data. However, the case of high-level neural coding[1] is different because redundancy already exists in nature and it's the brain's task to come up with an algorithm to exploit this redundancy and do error correction. (probably this is why we have training periods for brain). Now may be we can design decoders that do not know the structure of the coder and learn it as time goes by.

- In [5], [6] and similar works, what is mostly important is having good source codes being implemented by a population of neurons. In other words, we seek interconnection among neurons such that their output is uncorrelated without any redundancy among them. However, from a channel coding point of view, it is better to have some redundancy left. A simple way to verify this idea is to check the output of real population of neurons and see if there are any correlations in their output. If there is, it could be a sign of channel coding as well.

- In [3], authors have proposed a method to find the weight vector of an artificial BAM which results in networks with largest error correction

[1] By high-level I mean the methods used in brain to correct high-level errors such as the one made in a writing or hearing a sound. In contrast, low-level error correction refers to error correction at neuronal level.

ability. Extending this work to real neuronal networks is a simple step which brings us closer to analyze coding properties of neuronal networks. Coding theory could play an important role here as it can provide us with weight updating rules that results in the optimal weight vector after a few steps. This is in sharp contrast to the GA-based approach used in [3] as neurons simply can not perform GA in a neural network.

References

[1] P. Hester, J. J. Foxe, S. Molholm, M. Shpaner, H. Garavan, "Neural Mechanisms Involved in Error Processing: A Comparison of Errors Made With and Without Awareness", NeuroImage, Vol. 27, 2005, pp. 602-608.

[2] E. Doi, D. C. Balcan, M. S. Lewicki, "A Theoretical Analysis of Robust Coding over Noisy Overcomplete Channels", Advances in Neural Information Processing Systems, Vol. 18, 2006, pp. 307-314

[3] D. She, J. B. Cruz, "Encoding Strategy for Maximum Noise Tolerance Bidirectional Associative Memory", IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 16, NO. 2, MARCH 2005

[4] A. G. Dimitrov, J. P. Miller, "Neural Coding and Decoding: Communication Channels and Quantization", Network: Comput. Neural Syst., Vol. 12, 2001, pp. 441472.

[5] M. Aghagolzadeh, S. Eldawlatly, K. Oweiss, "Coding Stimulus Information with Cooperative Neural Populations", Proc. IEEE International Symposium on Information Theory (ISIT), 2009, pp. 1594-1598.

[6] D. H. Johnson, W. Ray, "Optimal Stimulus Coding by Neural Populations Using Rate Codes", Journal of Computational Neuroscience, Vol. 16, 2004, pp. 129138.

[7] D. H. Johnson, "Information Theory and Neuroscience: Why is the intersection so small?", IEEE Information Theory Workshop (ITW), 2008, pp. 104-108