Neural networks with Hebbian learning do not outperform random ones in fan-out systems

Marta García-Sánchez[1,2], Ramón Huerta[1,2]

[1] Institute for Nonlinear Science, University of California at San Diego La Jolla CA 92093-0402 USA

[2] GNB. Escuela Técnica Superior de Informática, Universidad Autónoma de Madrid, 28049 Madrid (SPAIN).

Abstract

There are many network structures in the brain that have a strong divergence of connections from one processing layer to the next. Good examples are found in the olfactory system of insects and in the Hippocampus. This paper elaborates on the advantages or disadvantages of using Hebbian learning in fan-out systems as opposed to sparse randomly connected networks. The measure to determine performance is information conservation from one layer to the next. We conclude that in fan-out systems Hebbian learning does not improve information conservation and increases the level of activity in the network.

1   Introduction

The olfactory system of insects has a very intriguing network organization. The information gathered by the sensors in the antenna is compressed into a few hundreds of neurons in the antennal lobe (AL) and then is expanded again into the mushroom body (MB)[1, 2]. Throughout this process the neural activity decreases resulting in sparse coding in the MB [3]. Moreover, it has been observed that the spatio-temporal activity in the MB is stimulus independent, *i.e.* for different odor concentrations the average activity remains nearly constant[4].

It is common belief that the first relay stations in olfactory information processing create a large coding space in which the representations of different classes of odors are spread out to allow easy discrimination [5]. In the general problem of linear classification Thomas Cover proved that nonlinear projections into a high dimensional space allows better classification [6]. It is also the inspiration of support vector machines [7] and it appears that the olfactory system of insects may follow this strategy. It has been experimentally observed that the MB is crucial for learning of odors[8, 9, 10]. However, the exact locations of the synapses involved in this process could not be determined. We intend by means of this brief communication to determine whether it is reasonable to locate the synaptic changes between the AL and the MB. The Dentate Gyrus in the hippocampus also has random sparse activity [11], which is thought to be better for associative memory [12]. Therefore, we believe that this study is also extensible to the fan-out in the hippocampus formations.

In a previous paper [13] we calculated the limits for the parameters of the fan-out phase that allow to maintain the information from one layer to the next while having a sparse code as experimentally observed. The main assumption of those calculations is a random connectivity from the AL to the MB. We believe that this assumption is not only convenient for the theoretical calculations but is also necessary given the available natural technology. It is also reasonable that the network is not specifically tuned to a given set of inputs, so it is able to equally represent any incoming input.

Our results show that if we apply Hebbian learning to the synapses the information representation is not improved. It remains close to the same values obtained with random connections. Moreover, the activity levels of the MB highly increase when Hebbian learning is used. Therefore, Hebbian learning does not outperform random connectivity in fan-out networks.

The organization of the paper is as follows. One section is about the model description and the measures used to determine the information conservation and activity levels. In the result section we show that the Hebbian learning does not improve the information representation. Finally, we discuss where Hebbian learning is more likely to be effective.

2   Model description

The kenyon cells (KC) of the MB are modeled with a McCulloch-Pitts neurons [14], whose equation are

$$y_j = \left\{ \begin{array}{ll} 1 & \text{if } \sum_{j=1}^{N_I} c_{ji} \cdot x_i \geq \theta, \\ 0 & \text{otherwise.} \end{array} \right\} \equiv F(\mathbf{x}) \tag{1}$$

where x_i and y_j are, respectively, the activities of the neurons in the AL and the KCs, and c_{ij} is the connectivity matrix. All these variables take binary values: for the activities, the spike event is represented by 1 and the rest state is given by 0, and in the connectivity matrix a value of 1 means that there is a synaptic connection from the ith AL neuron to the jth KC one, and a 0 means no connection. The fan-out system is characterized by the following parameters: N_I is the total number of neurons in the input layer, N_O is the number of neurons in the output state, p_I is the mean activity of an input neuron, θ is the firing threshold, and p_C is the mean connectivity value from the AL to the MB. For example, the *locust* has an average of 20 connections from the AL to a given KC, which results in a very sparse connectivity matrix.

2

3 Injectivity measure

The injectivity measure of the function $F(\mathbf{x})$ determines the quality of the transmission of data from one layer to the next one. The goal is to find out whether every different input produces a different output, that is, whether F is injective or one-to-one such that no information is lost during the process. The measure of injectivity we use is based on taking pairs of different inputs and comparing the outputs. The greater the number of different output pairs, the greater the injectivity. This calculation is done over many different systems and many input pairs. The *injectivity* measure can be formally expressed as

$$I(F) \;=\; \langle\langle P(\mathbf{y} \neq \mathbf{y}'|\mathbf{x} \neq \mathbf{x}', C)\rangle_{\mathbf{x}\neq\mathbf{x}'}\rangle_C \;=\; 1 - P_{col} \tag{2}$$

where P_{col} is the collision probability. We define a *collision* as the event of having two different input states (vectors) that are producing the same final state. For sparse random connections the collision probability is $P_{col} = \{p_O{}^2 + (1 - p_O)^2 + 2\sigma^2\}^{N_O} \simeq e^{-2N_O p_O}$ (see [13] for more details), where p_O is the mean output activity that for random matrices can be expressed as [13]

$$p_O = 1 - \sum_{i=1}^{\theta-1} \binom{N_I}{i} (p_C p_I)^i (1 - p_C p_I)^{N_I - i}.$$

4 Synaptic strength modification

Hebbian learning is the classical choice for local synaptic changes. The specific learning rule we use here is

$$c_{ij}(x_i, y_j) = \begin{cases} 1 & \text{with probability } p_+ \text{ if } x_j = 1 \text{ and } y_j = 1 \\ 0 & \text{with probability } p_- \text{ if } x_j = 0 \text{ and } y_j = 1 \\ \text{unchanged} & \text{if } x_j = 1 \text{ and } y_j = 0 \\ \text{unchanged} & \text{if } x_j = 0 \text{ and } y_j = 0 \end{cases} \tag{3}$$

A connection (synapse) is activated with probability p_+ if the input activity is accompanied by an output response. The connection is removed with probability p_- however if an output occurs in absence of input activity. In the remaining cases (no input and no output and input but no output) the synapse remains unaltered. Out of the many possible combinations of the input-output table we think that this is the most reasonable. In particular, the third rule, (for $x_j = 1, y_j = 0$) is left unchanged because the activity levels are much higher in the input than in the output and we deal with sparse code. The expected number of synaptic changes in $(x_j = 1, y_j = 0)$ would be quite high if we had a probability of change, p_-, i. e., approximately $N_I\,N_O\,p_I\,p_-$, while for $(x_j = 0, y_j = 1)$ is approximately $N_I\,N_O\,p_I\,p_-\,p_O$. It is also known that when there is not activity postsynaptically there are not synaptic strength changes.
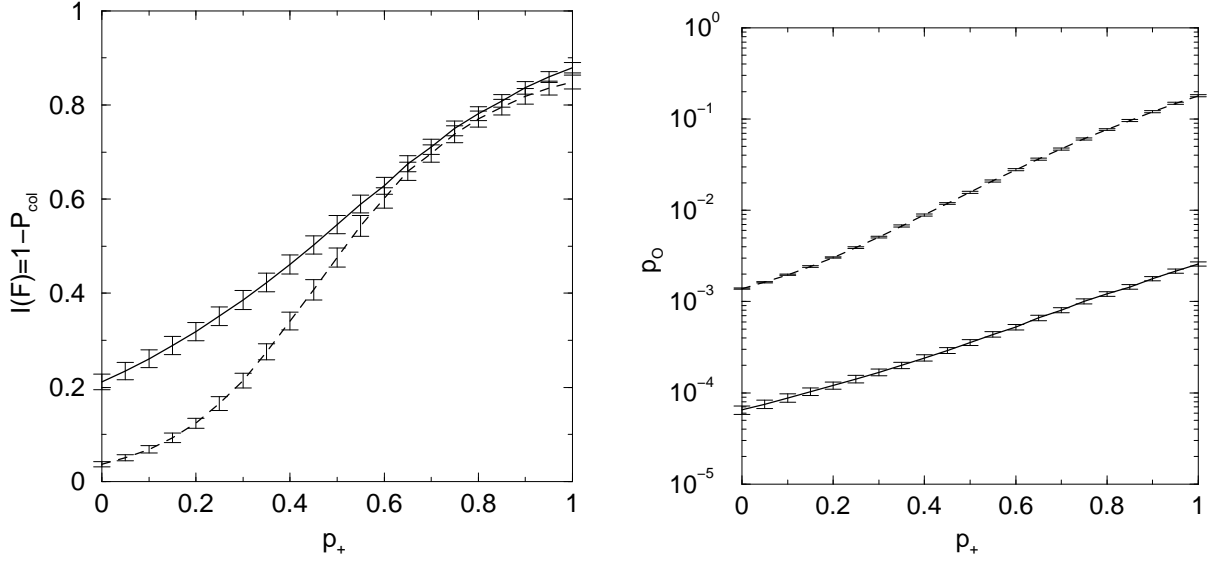
Figure 1: **Left**: Values of the injectivity measure for different learning rates for networks with Hebbian learning (dashed line) and random network (solid lines). Note that the random generated networks are independent of p_-. However for each p_- the Hebbian process generates a p'_C that is used to produce the random network. **Right**: Level of activity for different learning rates (dashed line) with their corresponding random network with probability $p'_C(p_+)$ (solid lines). The random networks always display a lower level of activity than the Hebbian ones.

5   Results

The simulation protocol is the following. One trial has two parts: first, we apply Hebbian learning during $50,000$ presentations of randomly chosen inputs generated by a Bernoulli process with probability $p_I = 0.1$; second, once the learning phase is finished, we estimate the probability of having a collision given different inputs, P_{col}. Then we run 500 trials, where each trial has a different random connectivity matrix, c_{ij}, each generated with probability $p_C = 0.1$. The probability of collision is then estimated using 5000 different couples of non-colliding inputs. The size of the input and output layer are taken from the *drosophila*, that is, $N_I = 100$ and $N_O = 2,500$.

The learning process is tested for several values of p_+ and p_-. After the learning process has completed, the average number of connections has changed to a new value, p'_C. We need to compare the random network and the network with Hebbian learning in the same terms. Therefore, we also generate a random network with the new value p'_C and we compare the injectivity and the average activity values.

4

Fig. 1 shows the injectivity values for different p_+. The solid curve that represents the random network has slightly better injectivity values than the Hebbian one. The error bars are just the mean standard deviation values calculated from all the trials. We can conclude that Hebbian learning does not produce a better representation than a random network. On the right side of the figure, we plot the activity levels in the fanout layer, $\mathbf{y}$, for different p_+ that produces different values of p'_C. One of the consequences of the specific tuning of the learning process is that the total level of activity is higher. Therefore, the sparse level is decreased for these networks. Another inconvenience if we keep in mind that there is sparse coding in the biological system.

6   Discussion

Our result indicates that Hebbian learning does not produce a better representation of the information than random connectivity matrices in the fan-out phase of the olfactory system of insects. This result is also extensible to the dentate gyrus in the hippocampus because of the sparse and random appearance of the neural activity.

The advantage of having a random connectivity matrix is that there is no specificity to any odor. The insect can smell a new odor and it will always have the same level of activity in the MB (as observed experimentally [4]). Therefore, it seems reasonable that the MB is not specifically tuned by Hebbian learning to any given set of inputs.

Now, the question that arises is how the sparsely and evenly represented information in the MB is used. The answer is classification for the olfactory system or associative memory for the hippocampus. Once we have well separated and easily identifiable projected inputs in the fan-out layer we can tune the system for classification purposes in the next processing layers by Hebbian learning. The next processing layer are the $\alpha, \beta, \gamma - lobes$ in the insects, and the CA3 and CA1 regions in the hippocampus. We will show in future work how we can tune the network to carry out a classification task using mutual inhibition and Hebbian learning [15] making use of the randomizing effect of the fan-out phase. The present work therefore predicts that the synaptic changes that are involved in learning process are the ones from the fan-out layer (the KC layer) to the α, β, γ -lobes.

Acknowledgments

References

[1] Laurent G (1999) A Systems Perspective on Early Olfactory Coding, Science 286: 723-728.

[2] Laurent G, Stopfer M, Friedrich RW, Rabinovich MI, Volkovskii A, Abarbanel HDI. (2001) Odor encoding as an active, dynamical process: Experiments, computation, and theory. Annual Review of Neuroscience 24:263-297.

[3] Perez-Orive J, Mazor O, Turner GC, Cassenaer S, Wilson RI, Laurent G. (2002) Oscillations and the Sparsening of Odor Representations in the Mushroom Body. 297: 359-365

[4] Wang YL, Wright NJD, Guo HF, Xie ZP, Svoboda K, Malinow R, Smith DP, Zhong Y (2001) Genetic manipulation of the odor-evoked distributed neural activity in the Drosophila mushroom body. Neuron 29(N1):267-276.

[5] Gilles L (2002) Olfactory network dynamics and the coding of multidimensional signals. Nature Reviews Neuroscience **3**, 884 - 895

[6] Cover, T. (1965). Geometric and statistical properties of systems of linear in-equalities with applications in pattern recognition. IEEE Tran. Elect. Comput., 14, 326.

[7] Cortes C, Vapnik V (1995) Support vector networks. Machine Learning 20: 273-297.

[8] McGuire SE, Le PT, Davis RL (2001) The role of Drosophila mushroom body signaling in olfactory memory. Science 293(N5533):1330-1333.

[9] Dubnau J, Grady L, Kitamoto T, Tully T (2001) Disruption of neurotransmission in Drosophila mushroom body blocks retrieval but not acquisition of memory. Nature 411(N6836):476-480.

[10] Zars T, Fischer M, Schulz R, Heisemberg M (2000) Localization of a short-term memory in Drosophila. Science 288:672-675.

[11] Barnes CA, McNaughton BL, Mizumori SJ, Leonard BW, Lin LH. Comparison of spatial and temporal characteristics of neuronal activity in sequential stages of hippocampal processing. Prog Brain Res 1990;83:287-300

[12] Randall C. O'Reilly, Kenneth A. Norman Hippocampal and neocortical contributions to memory: advances in the complementary learning systems framework, Trend in Cognitive Sciences **6** (12) 2002.

[13] Garcia-Sanchez M, Huerta R Journal of Computational Neuroscience (in press).

[14] McCulloch WS, Pitts W (1943) A Logical Calculus of Ideas Imma- nent in Nervous Activity, Bulletin of Mathematical Biophysics 5: 115-133.

[15] R Huerta, T. Nowotny, M. García-Sanchez, H. D. I. Abarbanel, M. I. Rabinovich, The classification problem in the olfactory system of insects (in preparation).