Finding structure by entropy minimization in coupled reconstruction networks

Szatmáry Botond [a] Barnabás Póczos [a] András Lőrincz [a,1]

[a] *Department of Information Systems*
Eötvös Loránd University
Pázmány Péter sétány 1/C., 1117 Budapest, Hungary

Abstract

There is psychological and physiological evidence for components-based representations in the brain. We present a special architecture of coupled parallel working reconstruction subnetworks that can learn components of input and extract the structure of these components. Each subnetwork directly minimizes the reconstruction error and indirectly minimizes the entropy of the internal representation via a novel tuning method, which effectively reduces the search space by changing the learning rate dynamically and increasing the escape probability from local minima. Revealing the structure of the input improves when competitive spiking and indirect minimization of the entropy of spike rate are applied *together*.

Key words: grouping components, reconstructive network, spike code, rate code, indirect entropy minimization, dynamic learning rate

1 Introduction

In this paper an architecture comprising reconstruction neural networks is introduced. The architecure demonstrates the appealing joined features of spike coding and rate coding (see [1] for an overview on neuronal coding). The subject of our work is to find efficient and biologically plausible mechanisms that search for components and structure in the input data and are able to develop efficient sparse representation.

Our motivation comes from recent theoretical models on neocortical processing (see, e.g., [2]) which claim that sensory processing may be based on recon-

[1] Corresponding author. e-mail: lorincz@inf.elte.hu

struction networks in which temporal integration takes place at the internal (hidden) representation. Moreover, we are also considering that (i) layer V of neocortical sensory processing areas, the putative container of the internal representation, influences control structures at some stage of the development [3] and that (ii) information processing has long delays. Thus, it follows that internal representation of memory components should not change often in order to maintain smooth and continuous controlling. Another hint of component formation is provided by a theory of human visual recognition, claiming that recognition works through the recognition of complementing components [4]. Therefore, to discover such complementing components, we have applied non-negativity constraint [5] both on neural activities and on synaptic weights.

In our model, information is represented by series of spike sets, spikes are selected by a stochastic winner take all (WTA) mechanism, whereas rate code –the averaging of spikes in a moving window of time– forms the internal representation. Non-negativity constraints enables to use entropic prior [6] on the internal representation, which prior biases the searches towards components whose activity (in the internal representation) can survive spike rate averaging and, therefore, are smooth.

2 Architecture

The full architecture is made of K parallel working subnetworks. All of them generate their $\mathbf{y}^{(i)} \in \mathbf{R}_+^n$ reconstruction vectors independently. $\mathbf{R}_+$ is the set of non-negative real numbers, superscript $i \in \{1, \ldots, K\}$ denotes the i^{th} subnetwork, $\mathbf{y}^{(i)}$ is the reconstruction vector of the i^{th} subnetwork. The common reconstruction process is the only coupling between the subnetworks: The input $\mathbf{x} \in \mathbf{R}_+^n$ is approximated (reconstructed) by the internal (hidden) representations $\mathbf{h}^{(i)} \in \mathbf{R}_+^r$ and the top-down (TD) generative transformation matrices $\mathbf{W}^{(i)} \in \mathbf{R}_+^{n \times r}$: $\mathbf{x} \approx \mathbf{y} = \sum_{i=1}^K \mathbf{y}^{(i)} = \sum_{i=1}^K \mathbf{W}^{(i)} \mathbf{h}^{(i)}$, where $\mathbf{y} \in \mathbf{R}_+^n$ is the common reconstruction vector of all subnetworks. For simplicity, the transposed forms ($\mathbf{W}^{(i)^T}$) of the same matrices are used for bottom-up (BU) transformations. The r columns of TD matrices $\mathbf{W}^{(i)}$ contain the generative weights and are called basis vectors or memory components. Each subsystem processes the common reconstruction error $\mathbf{e} = \mathbf{x} - \mathbf{y}$. Figure 1 depicts the architecture.

For each subsystem, the BU processed reconstruction error, $\mathbf{s}^{(i)} = \mathbf{W}^{(i)^T} \mathbf{e}$ ($\in \mathbf{R}^r$), undergoes non-linear transformation: $\mathbf{u}^{(i)} = f_{WTA}(\mathbf{s}^{(i)})$ ($\in \mathbf{R}^r$), where $f_{WTA}(.)$ denotes the stochastic WTA function and each $\mathbf{u}^{(i)}$ is an r dimensional unit vector. Function $f_{WTA}(.)$ selects a 'firing unit' from the exponentiated and

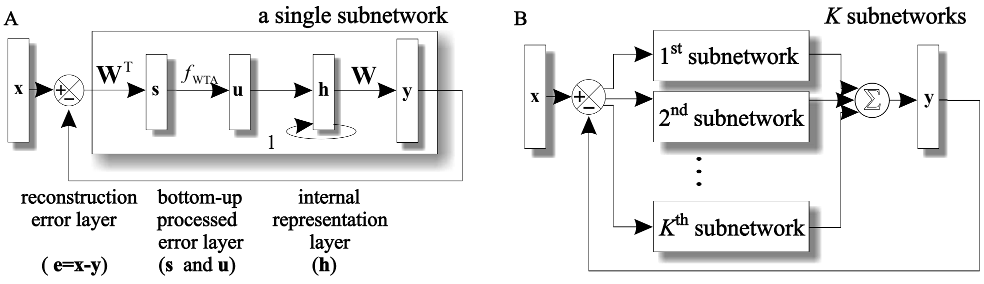

Figure 1. **Illustration of the architecture. A:** A basic reconstruction network unit (superscripts not shown). **B:** The architecture is made of K parallel working basic units (subnetworks), having a single reconstruction error layer, which is the only coupling between the networks. To each input, the full network iterates until convergence or reaching a maximum iteration number.

normalized probability distribution created from the components of vector $\mathbf{s}^{(i)}$:

$$p_j^{(i)} = \frac{\mathrm{e}^{\vartheta\,(\mathbf{W}^{(i)^T}(\mathbf{x}-\mathbf{y}))_j}}{\sum_k \mathrm{e}^{\vartheta\,(\mathbf{W}^{(i)^T}(\mathbf{x}-\mathbf{y}))_k}} = \frac{\mathrm{e}^{\vartheta\, s_j^{(i)}}}{\sum_k \mathrm{e}^{\vartheta\, s_k^{(i)}}} \quad i=(1,\ldots,K),\; j=(1,\ldots,r), \quad (1)$$

where parameter $\vartheta > 0$ can shape probability distribution $p^{(i)} = \prod_j p_j^{(i)}$. The new values of the $\mathbf{h}^{(i)}$ internal representation vectors are computed by moving window averaging of $f_{WTA}(.)$ outputs that corresponds to temporal integration with exponential kernel in the infinitesimal time step limit:

$$\mathbf{h}^{(i)} \leftarrow (1-\alpha)\,\mathbf{h}^{(i)} + \alpha\,\mathbf{u}^{(i)} \quad i=(1,\ldots,K) \quad (2)$$

The rate code of $\mathbf{u}^{(i)}$ (i.e., $\mathbf{h}^{(i)}$) is influenced by the length of the moving window proportional to $1/\alpha$, where $0<\alpha<1$.

For the approximation of input $\mathbf{x}$ and for deriving learning rules for matrices $\mathbf{W}^{(i)}$, a cost function is defined. Sparse Cauchy probability distribution is assumed for the reconstruction error and an entropic prior [6] is used to bias the choice of the internal representation vectors:

$$P\left(\mathbf{x},\,\mathbf{h}^{(1)},\ldots,\mathbf{h}^{(K)} \mid \mathbf{W}^{(1)},\ldots,\mathbf{W}^{(K)}\right) \sim \frac{1}{1+\|\mathbf{x}-\mathbf{y}\|^2}\,\mathrm{e}^{\sum_{i=1}^{K} -\kappa H\left(\mathbf{h}^{(i)}\right)},$$

where $\kappa > 0$ is a constant and $H(\cdot)$ denotes the discrete entropy function. P is the probability distribution that represents our choices and has to be maximized or, its inverse –which we shall take as our cost function– has to be minimized:

$$J_{\mathbf{W}^{(i)}} = (\|\mathbf{x} - \textstyle\sum_{j=1}^{K} \mathbf{W}^{(j)}\mathbf{h}^{(j)}\|^2)\,\mathrm{e}^{\sum_{i=1}^{K} \kappa H\left(\mathbf{h}^{(i)}\right)} \quad i=(1,\ldots,K) \quad (3)$$

Note that the additional constant 1 has been removed for convenience, but this change has no effect on the minimum of $\mathbf{W}^{(i)}$. Gradient descent provides the following update rule for matrix $\mathbf{W}^{(i)}$:

$$\Delta\mathbf{W}^{(i)} = \gamma\, \mathrm{e}^{\kappa\, H\left(\mathbf{h}^{(i)}\right)} \left(\mathbf{x} - \textstyle\sum_{j=1}^{K} \mathbf{W}^{(j)}\mathbf{h}^{(j)}\right) \mathbf{h}^{(i)T} \quad i = (1, \ldots, K), \tag{4}$$

where γ is a constant and corresponds to the usual learning rate. After each upgrade, each $\mathbf{W}^{(i)}$ is rectified by applying the non-linearity $W_{kl}^{(i)} = \max(0,\, W_{kl}^{(i)})$ to enforce the non-negativity constraint. The *effective* learning rate depends on γ *and* the entropy of the internal representation: $\gamma\, \mathrm{e}^{\kappa\, H\left(\mathbf{h}^{(i)}\right)}$. The entropy modulation is influenced by parameter κ. Matrix $\mathbf{W}^{(i)}$ may undergo large modifications for large entropies but fine tuning occurs only when the entropy is small. In turn, the update will facilitate the network to *escape* from regions of high entropy internal representations and minimization of the entropy of the internal representation is indirectly comprised into the learning rule of $\mathbf{W}^{(i)}$: prior knowledge does not affect the direction of tuning but increases or decreases the learning rate dynamically, hereby increases or decreases the exploration time of different domains of the state space. Note that in each subnetwork only the corresponding entropy term modulates learning and, in turn, modulation is local: A subnetwork of low entropy may stay stable while others of high entropy may undergo learning.

Contrary to other parts of the learning rule, neuronal representation that could enforce the entropic prior by Hebbian means is not seen at this moment. However, one might assume that entropic prior is a local effect, acting through, for example, local chemical processes. Then entropic prior acts slowly, which is in full agreement with our rate code assumption that develops the prior. The conjecture that firing of two nearby neurons makes learning rates higher than the same amount of firing but from a single neuron, could be validated or falsified experimentally.

3 Computer simulations

We illustrate the component and structure finding ability of the architecture on the following problem: every input $\mathbf{x}$ represented an '8×8 image' consisting of two randomly chosen vertical and two randomly chosen horizontal bars. At overlapping bars, pixel values were added. Two subnetworks ($K = 2$) were used to reconstruct and to learn the bar structure. The corresponding TD matrices are denoted by $\mathbf{W}^{(1)}$ and $\mathbf{W}^{(2)}$. Clearly, the sparsest representation is gained if one of the TD matrices represents the horizontal bars and the other represents the vertical bars.

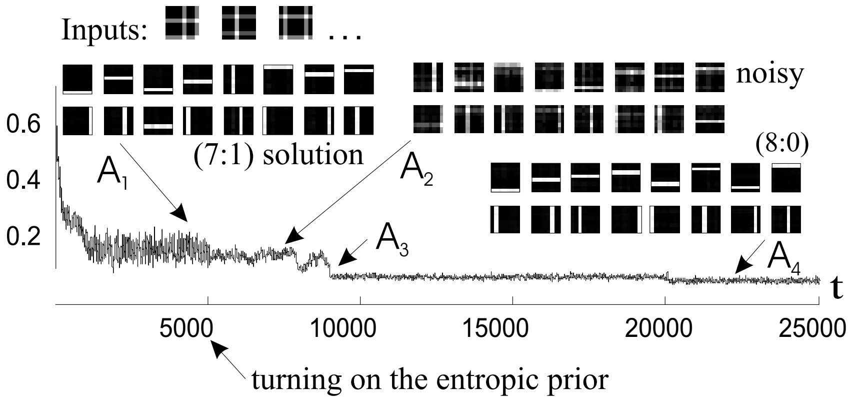

Figure 2. **Effect of parameters on learning:** Reconstruction error (vertical axis) vs. input number (t) (horizontal axis) of a demonstrative computer run. The inputs consisted of vertical and horizontal bars. The insets are examples of the developed TD matrices $\mathbf{W}^{(1)}$ and $\mathbf{W}^{(2)}$, belonging to the marked position. The upper (lower) row of an inset: memory components of the TD matrix of the first (second) subsystem. Notation: (n:m) denotes a solution with n horizontal (vertical) and m vertical (horizontal) bars in the first (second) TD matrix. Global minima are denoted by (8:0). Number of inputs: 25,000. The iteration number for each input was 70. In the first 5,000 input: $\kappa = 0$ (Eq. 4), i.e., the effect of entropy was neglected. At the $5,001^{th}$ input: entropic prior was turned on, κ was set to 2. At the $20,001^{th}$ input: κ was set to 0.8 that lowered the learning rate. A_1: typical results without entropic prior. A_2: escaping from the local minimum upon turning on the entropic prior. A_3: sudden finding of a global minimum. A_4: improved results for lower κ value.

At first, the effect of the entropic prior was neglected ($\kappa = 0$ in Eq. 4) and the probability distributions $p^{(i)}$ ($i = (1,2)$, Eq. 1) approximated a perfect winner take all mechanism with $\vartheta = 20$. In this case, learning converged to the global minimum in approximately 20% of the experiments, but was trapped in local minima in 80% of the cases. Local minima corresponded to (7:1) or to (6:2) solutions, that is the basis vectors of one TD matrix approximated 7 or 6 horizontal (or vertical) and 1 or 2 vertical (or horizontal) bars (e.g., inset A_1 of Fig. 2). Since the entropy of the internal representations close to a global minimum ((8:0) in this case) is small and it is higher around local minima (e.g. (7:1) or (6:2)), the situation has changed considerably when the entropic prior was turned on ($\kappa > 0$, at $t = 5000$ in Fig. 2). In this case, escapes from the local minima were successful and a global minimum, an (8:0) solution was found (mark A_3 and inset A_4 in Fig. 2) in 100% of the experiments. Thus, the entropic prior makes relevant contribution in the discovery of structures. In order to improve the basis vectors, we decreased the learning rate gradually by lowering the value of κ or γ of Eq. 4 (inset A_4 of Fig. 2).

In cases when the internal representation was not derived with spiking and averaging (Eq. 2), but continuous gradient method $\Delta\mathbf{h}^{(i)} = \mathbf{W}^{(i)^T}\left(\mathbf{x} - \sum_{j=1}^{K}\mathbf{W}^{(j)}\mathbf{h}^{(j)}\right)$ was applied, only (4:4) and (5:3) solutions were found under the same conditions.

4 Discussion

A reconstruction network architecture has been proposed for structure finding. It was shown that a coupled set of subnetworks is able to discover and group components of the input and generate the lowest entropy code. In a sense, we were looking for computational advantages of columnar organization, which, in our presumption, may help to overcome combinatorial explosion. Further, we assume that columnar organization is responsible for the discovery of components that serves the 'recognition by components' process [4]. Our observation is that an entropic prior acting locally and allowing considerably different learning rates in different columns satisfies the algorithmic requirements. Another intriguing finding is that both spiking operation and the rate code are necessary for finding structured components: The indirect entropy minimization of the internal representation biased the search towards components whose activity can survive the time window of rate code. Such long-lived components could not be found without spiking. Further, because the forefront of a spike train may properly approximate the relaxed representation, the network approximates fast feedforward signal transmission. Beyond the demonstration of the synergy of spike and rate codes, a novel contribution of our work is the indirect minimization of the cost function. Indirect minimization occurred by dynamically modulating the learning rate and that modulation was subject to the prior knowledge. Different modulations for different columns are possible.

References

[1] W. Gerstner, W. Kistler, Spiking Neuron Models, Single Neurons, Populations, Plasticity, Cambridge University Press, 2002.

[2] A. Lőrincz, B. Szatmáry, G. Szirtes, Mystery of structure and function of sensory processing areas of the neocortex: A resolution, Journal of Computational Neuroscience 13 (2002) 187–205.

[3] I. Diamond, Progress in Psychobiology and Physiological Psychology, Academic Press, New York, 1979, Ch. The subdivision of neocortex: A proposal to revise the traditional view of sensory, motor and association areas, pp. 1–43.

[4] I. Biederman, Recognition-by-components: A theory of human image understanding, Psychological Review 94 (1987) 115–147.

[5] D. Lee, H. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (1999) 788–791.

[6] M. Brand, Pattern discovery via entropy minimization (1998).
URL http://citeseer.nj.nec.com/brand98pattern.html