Boolean Factor Analysis by Hopfield-Like Autoassociative Memory

Frolov, A. A.
2006

Institute of Computer Science
Academy of Sciences of the Czech Republic

Boolean factor analysis by Hopfield-like autoassociative memory

A. A. Frolov and D. Húsek and I. P. Muraviev and P. A. Polyakov

Technical report No. 961

February 2006

Pod Vodárenskou věží 2, 182 07 Prague 8, phone: (+420) 26605 3230, fax: +420 286 585 789,
e-mail:dusan@cs.cas.cz

Institute of Computer Science
Academy of Sciences of the Czech Republic

Boolean factor analysis by Hopfield-like autoassociative memory [1]

A. A. Frolov[2] and D. Húsek[3] and I. P. Muraviev[4] and P. A. Polyakov[5]

Technical report No. 961

February 2006

Abstract:

The feature space transformation is a widely used method for data compression. Due to this transformation the original patterns are mapped into the space of features or factors of reduced dimensionality. In this paper we demonstrate that Hebbian learning in Hopfield-like neural network is a natural procedure for Boolean factor analysis. Due to this learning, neurons that tend to fire together (represent one common factor) are more correlated and thus create an attractor of the network dynamics. If the attraction basins around factors are large enough, the factors could be revealed by random search. This paper is dedicated to estimation of the size of attraction basins around factors. Two global spurious attractors are shown to prevent convergence of the network activity to the factors invalidating any procedure of their search. These global attractors can be completely deleted from network dynamics by introducing a single inhibitory neuron with bi-directional Hebbian synapses. Due to additional inhibition, the size of attraction basins around factors becomes the same as around the stored patterns in usual Hopfield network. The procedure of factors search is described in the accompanying paper.

Keywords:
Boolean factor analysis, Hopfield neural network, unsupervised learning

[1]The work was partly supported by the Institutional Research Plan AV0Z10300504 "Computer Science for the Information Society: Models, Algorithms, Appplications" and by grant Intelligent methods for increasing of reliability of electrical networks 1ET100300414 granted by GA AS CR

[2]Institute of Higher Nervous Activity and Neurophysiology of the Russian Academy of Sciences, Butlerova 5a, 117 485 Moscow, Russia; e-mail: aafrolov@mail.ru

[3]Institute of Computer Science Academy of Science of the Czech Republic, Pod Vodárenskou věží 2, 182 07 Prague 8, Czech Republic; tel. (+420)26605 3230, Fax: (+420) 28658 5789, e-mail: dusan@cs.cas.cz

[4]Institute of Higher Nervous Activity and Neurophysiology of the Russian Academy of Sciences, Butlerova 5a, 117 485 Moscow, Russia;e-mail: muravevi@mail.ru

[5]Institute of Optical Neural Technologies of the Russian Academy of Sciences, Vavilova 44, 119 333 Moscow, Russia; e-mail: pavel@8ka.mipt.ru

1 Introduction

Factor analysis is one of the most efficient methods to reveal and to overcome informational redundancy of high-dimensional signals. Factors extraction is a procedure which maps original signals into the space of factors. The principal component analysis (PCA) is a classical example of such mapping in the linear case. Linear factor analysis implies that each original signal can be presented as

$$\mathbf{X} = \mathbf{FS} \tag{1.1}$$

where $\mathbf{F}$ is a matrix $N \times L$ of factor loadings and $\mathbf{S}$ is a vector of factor scores. The columns of $\mathbf{F}$ represent factors in the original signal space. Each component of $\mathbf{S}$ gives contribution of a corresponding factor in the original signal. The mapping of the original space to the factor space means that signals are represented by vectors $\mathbf{S}$ instead of original vectors $\mathbf{X}$. Informational redundancy of original signals is reduced if dimensionality L of vectors $\mathbf{S}$ (the number of factors) is smaller than the dimensionality N of the original signals.

PCA suggests that factor scores should be statistically independent and the vectors of factor loadings orthogonal. Then the vectors of factor loadings are eigenvectors of the covariation matrix $\mathbf{J} = \mathcal{M}\{\mathbf{X}\mathbf{X}^{\mathbf{T}}\}$ where $\mathbf{X}^T$ is transposed $\mathbf{X}$. Dispersions of factor scores are eigenvalues of covariation matrix. Eigenvector $\mathbf{f}^1$ with the highest eigenvalue Λ_1 (factor with the highest contribution to the total variance of signals $\mathbf{X}$) can be easily obtained [19] by the iterative procedure

$$\mathbf{X}(t+1) = N(\mathbf{h}(t)) \tag{1.2}$$

where

$$\mathbf{h}(t) = \mathbf{JX}(\mathbf{t}) \tag{1.3}$$

and $N(\mathbf{h}) = \mathbf{h}/|\mathbf{h}|$ denotes vector normalization. Starting from random initial vector $\mathbf{X}_{in}$, $\mathbf{X}(t)$ tends to $\mathbf{f}^1$. It is easy to show [19] that during this iterative procedure, Lyapunov function $\Lambda = \mathbf{X}^T\mathbf{JX}$ monotonically increases and reaches Λ_1. When $\mathbf{f}^1$ is obtained, the iterative procedure can be applied to matrix $\mathbf{J} - \Lambda_1\mathbf{f}^1\mathbf{f}^{1T}$ to obtain the next eigenvector of matrix $\mathbf{J}$, and so on.

This procedure can be obviously described in terms of a neural network approach. Covariation matrix $\mathbf{J}$ corresponds to a matrix of synaptic connections obtained by Hebbian learning. The iterative procedure corresponds to the evolution of activity in neural network with parallel dynamics where $\mathbf{h}$ is a vector of synaptic excitations. And substraction of the found factor from a covariation matrix corresponds to Hebbian unlearning. Linear and even some nonlinear PCA procedures have been actually realized by the neural network approach [12], [18], but only for special cases of nonlinearity.

One particular form of nonlinear factor analysis is a binary one, where a complex vector signal (pattern) has a form of the Boolean sum of weighted binary factors:

$$\mathbf{X} = \bigvee S_l\mathbf{f}^l. \tag{1.4}$$

In this case, original signals, factor scores and factor loadings are binary. In contrast to linear factor analysis, the dimensionality L of vector $\mathbf{S}$ (number of factors) can be larger than dimensionality N of original pattern space. If the mean number C of factors mixed in each original signal $\mathbf{X}$ (we treat C as signal "complexity") is much smaller than the total number of factors, then a large reduction of informational redundancy can be achieved even in this case. To show it let us assume that the components of $\mathbf{X}$ are statistically independent, i.e ignore that signals can be presented in form (1.4). Then $I_S = NH(q)$ bins of information are required to represent original signal, where $H(q) = -q\log_2 q - (1-q)\log_2(1-q)$ is a Shannon entropy function and q is a probability that a given component of $\mathbf{X}$ is equal to one. The mapping of the original space to the factor space means that signals are represented by binary vectors $\mathbf{S}$ instead of original vectors $\mathbf{X}$. Since vector $\mathbf{S}$ of dimensionality L contains C ones, its representation requires $I_F = LH(C/L)$ bins of information. Reduction of information redundancy is achieved if $I_F < I_S$.

There is a few implemented methods for Boolean factor analysis. However, these methods [6],[16],[3] are time consuming and do not support large data sets. Thus their applicability is limited only to the

case of relatively small dimensionality. Our new Neural Network attempt should bring an innovative way how to handle large(parallel computation) dynamically changing data (incremental learning). It was a challenge for us to utilize the Hopfield-like neural network with parallel dynamics for Boolean factor analysis because it has a lot of similarities with the iterative procedure described above for linear factor analysis. First, the connection matrix of this network is a covariation matrix of input signals obtained by Hebbian learning. Second, its activity is determined by the same iterative procedure (1.2, 1.3) except that normalization of the vector of synaptic excitations $\mathbf{h}$ is replaced of its binarization. And third, its activity has almost the same Lyapunov function

$$\Lambda(t+1) = \mathbf{X}^T(t+1)\mathbf{J}\mathbf{X}(\mathbf{t}). \tag{1.5}$$

For the Hopfield network, the formula for Lyapunov function slightly differs from that of linear case because the activity of the Hopfield-like network with parallel dynamics converges not only to point attractors but also to cyclic attractors of length two [10]. Respectively, $\mathbf{X}^T(t)$ in the formula for linear case must be replaced by $\mathbf{X}^T(t+1)$ for the binary case.

Since the neurons that represent one common factor tend to fire together Hebbian learning provides tighter connections between these neurons than between neurons belonging to different factors. Therefore, factors create attractors of the network dynamics similarly to eigenvectors of the correlational matrix in iterative procedure for linear case. However, the Hopfield-like network has one principal peculiarity. The network dynamics converges to one of the factors only when the initial state falls inside its attraction basin. Otherwise it converges to one of the spurious attractors. Note that for linear case it converges to one of the factors starting from any random initial state. Thus, two main questions arise in view of binary factor analysis by the Hopfield-like network. First, how often would the network activity converge to one of the factors starting from the random state? Second, is it possible to distinguish true and spurious attractors when the network activity converges to some point or cyclic attractor?

The probability of the network activity to converge to one of the factors depends evidently on the size of their attraction basins. Generally, this size is determined by three network parameters: complexity of input signals C, a relative number of factors L/N and sparseness of factors encoding. Sparseness is determined by the ratio $p = n/N$ of active neurons n in the factor to the network size N. Only sparse encoding is considered in the present paper. There are two a priori reasons to restrict the analysis by this case. First, Boolean superposition of even a relatively small number of densely encoded factors results in presentation of input signals as binary vectors composed of almost only ones. Since in the extreme case, when they composed of only ones, decomposition of signals in factors is evidently impossible, it is difficult to expect successful factorization of densely encoded factors. Second, as shown in [8] for ordinary Hopfield network, the size of the attraction basins around the stored prototypes is the largest when p has an order of 10^{-2}. Note that the ordinary Hopfield network corresponds to the extreme case of the considered network when its complexity C becomes one. It is reasonable to expect that the properties of networks with different complexities but with the same sparseness are close. Thus, sparse encoding of factors with p of the order of 10^{-2} seems to be the most preferable to provide the largest attraction basins around the factors. In the present paper most computer simulations were performed for $p = 0.02$.

The main goal of the present paper is to answer the question "What are the parameters of the system that guarantee existence of attractors of network activity corresponding to factors?". To answer this question we investigated the conditions under which the factors actually form attractors in the Hopfield-like neural network and estimated the size of attraction basins around them. The procedure for factors search is suggested in the accompanying paper.

The analysis is performed by Single Step (SS) approximation [14] and by computer simulation. SS is known to be rather inaccurate for the densely encoded Hopfield network and more sophisticated methods, such as the method of Statistical Neurodynamics (SN), are usually recommended. SN was elaborated initially for densely encoded network [1] and then modified for sparse encoding [17], [8]. However, as shown in [9], contrary to dense encoding, for sparse encoding (p is of the order of 10^{-2})the SN is even less accurate than the SS. Thus the SS seems to be more preferable for the present study.

However computer simulation revealed failure of SS approximation for Boolean factor analysis due to the dominance of two global spurious attractors for $C > 10$ that prevent convergence of network

activity to factors. These two attractors can be completely excluded from network dynamics by addition to the network of a single inhibitory neuron with bi-directional Hebbian synapses. As a result, the size of attraction basins around the factors greatly increases and becomes close and even slightly larger than the SS predicts.

The paper is organized as follows. A formal model description is given in Section 2. Section 3 exposes results of SS approximation. The properties of multi-step retrieval is described in Section 4. Section 5 is a short general discussion.

2 Network description

The neural network under consideration consists of N neurons of the McCulloch-Pitts type (integrate-and-fire binary neurons) with gradually ranged synaptic connections between them. Only a fully connected case is considered here.

Network is trained by a set of M patterns of the form $\mathbf{X}^m = \bigvee_{l=1}^{L} S_l^m \mathbf{f}^l$, where $\mathbf{f}^l \in B_n^N$ [6] are L factors (N dimensional binary vectors) and for every m-th pattern $\mathbf{S}^m \in B_C^L$ is a corresponding vector of factor binary scores. As follows from the definition every factor contains exactly $n = Np$ 1-s. Every complex pattern $\mathbf{X}^m$ contains, in turn, exactly C factors. We assumed factors and factor scores to be statistically independent. In a limit case when $C = 1$ the patterns become pure factors and we obtain an ordinary Hopfield case. In the opposite limit case $C = L$ all patterns of the learning set are identical (contain all factors) and, evidently factors, cannot be identified separately.

Connection matrix $\mathbf{J}$ is formed by using the correlational Hebbian rule:

$$J_{ij} = \sum_{m=1}^{M} (X_i^m - q^m)(X_j^m - q^m), \ i \neq j, \qquad J_{ii} = 0, \tag{2.1}$$

where bias $q^m = \sum_{i=1}^{N} X_i^m / N$ is the total activity of the m-th pattern. For $C = 1$ such a form of bias was shown to give the best informational properties [5]. This form of bias corresponds to the biologically plausible global inhibition being proportional to overall neuronal activity.

During the recall stage, on presentation of an initial pattern $\mathbf{X}^{in}$, the network activity evolves until it stabilizes in an attractor. As initial patterns we used distorted versions of factors with the same level of activity $n = Np$ as factors.

Evolution of the network activity in discrete time is determined by the synchronous dynamics equation for activity vector $\mathbf{X}$ at each time step:

$$\begin{aligned} X_i(t+1) &= \Theta(h_i(t) - T(t)), \ i = 1, \cdots N, \\ X_i(0) &= X_i^{in} \end{aligned} \tag{2.2}$$

where

$$h_i(t) = \sum_{j=1}^{N} J_{ij} X_j(t) \tag{2.3}$$

is synaptic excitation, Θ - step function, and $T(t)$ - activation threshold. The threshold $T(t)$ is chosen at each time step in such a way that the level of the network activity is kept constant and equal to n. Thus, on each step n "winners" (neurons with the greatest synaptic excitation) are chosen and only they are active on the next step. To avoid uncertainty in the choice of winners when several neurons have synaptic excitations at the level of the activation threshold, small random noise was added to the activation threshold of each individual neuron. The amplitude of the noise was put to be less than

[6] $B_n^N = \{\mathbf{X} | X_i \in \{0,1\}, \sum_{i=1}^{N} X_i = n\}$

the smallest increment of the synaptic excitation given by formula (2.3). This ensures that neurons with the highest excitations are kept to be winners in spite of the random noise being added to the neurons' thresholds. The noise around the thresholds of individual neurons was fixed during the whole recall process to provide its convergence. Since the number of active neurons in each factor is fixed and also equal to n, the choice of activation thresholds allows stabilization of the network activity in the vicinity of one of the factors. Thus, the type of the factors coding is fitted to the used recall procedure which allows avoiding explicit control of the activation threshold. As in the case when the activation threshold is fixed [10] only two types of attractors (point or cyclic of length two) are present in the network dynamics (see Appendix 1). The stable pattern (point attractor) or the first pattern of the cyclic attractor was taken in computer simulations as a resulting pattern (further termed as final pattern $\mathbf{X}^f$) of the recall process. In respect to the factor analysis problem each factor must have a corresponding stable pattern in its vicinity. If the network can recall all factors encoded in the complex training set, one can say that the factor analysis problem is solved successfully.

For the analysis of informational and dynamic properties of the network some integral parameters are introduced. Similarity between two vectors is measured in the Hamming space by their overlap m:

$$m(\mathbf{X}^1, \mathbf{X}^2) = \frac{1}{Np(1-p)} \sum_{i=1}^{N} (X_i^1 - p) X_i^2$$

In the case of vectors coincidence $m = 1$. If $\mathbf{X}^1 \in B_n^N$ and $\mathbf{X}^2$ is random and independent of $\mathbf{X}^1$, then their mean overlap is equal to zero. The overlap between initial state and the recalled factor is given by the overlap $m_{in} = m(\mathbf{f}^l, \mathbf{X}(0))$. The size of the attraction basin around the factors is a critical initial overlap m_{ab} which separates retrieval and not retrieval trajectories of neurodynamics.

As a measure of the relative informational loading we use $\alpha = LH(p)/N$, where $H(p)$ is the Shannon function which takes account of the sparseness level.

The network factor analysis ability is analyzed in dependence on the following five parameters: p, α, C, N, M under conditions $\mu = C^2/L$ and pC are of the order of 1 and $C/L \ll 1$. The size of the training set should be large enough so that each factor could be presented several times in combinations with different other factors. Therefore, we put $MC/L \gg 1$. Additionally we put $L \gg 1$, $N \gg 1$ and α is of the order of 10^{-1}.

3 Single-Step approximation

Single-step (SS) approximation has been proposed by Kinzel [14] for the densely encoded Hopfield network. It has been shown by other theoretical approaches [2], [1] and by Monte-Carlo simulations [11], [2], [15] that single-step approximation is very inaccurate for dense coding. However, it becomes quite accurate when sparseness increases [8].

The principal peculiarity of this approach is that at each time step one ignores the statistical dependence between the network activity and the connection matrix and takes account of only two macroparameters of neurodynamics: the overlap $m(t)$ between the current and recalled patterns and the total network activity. Since we have assumed that the network activity is constant at each time step and equal to the activities of factors, the recall process in our case is described by the evolution of only one parameter $m(t)$. Omission of the statistical dependence between the network activity and the connection matrix is possible only for the first step when the initial activity is actually stated independently of the connection matrix. This is why this approximation is called the "single-step" or "first- step" approximation.

Without any loss of generality, we may assume that a factor $\mathbf{f}^1$ is retrieved. In conformity with [4], we call neurons which are active and nonactive in $\mathbf{f}^1$ as "high' and "low" neurons, respectively. According to (2.1) and (2.3), the neurons synaptic excitations at the first step of the recall process can be presented in the form $h_i = \Sigma_i^1 + \Sigma_i^0$ where

$$\Sigma_i^\mu = \sum_{m \in \{m: S_1^m = \mu\}} (X_i^m - q^m) \sum_{j \neq i} (X_j^m - q^m) X_j^{in}$$

The first sum Σ_i^1 contains M^1 patterns of the training set which include the recalled factor $\mathbf{f}^1$ and Σ_i^0 contains M^0 patterns which do not include it. Since the variance of q^m is of the order $1/N$ (see Appendix 2), one can put

$$q^m = \langle q^m \rangle = q = 1 - (1-p)^C \simeq 1 - \exp(-pC) \tag{3.1}$$

where $\langle \cdots \rangle$ means averaging over all factor scores and all factors except $\mathbf{f}^1$. Each initial pattern includes $n_1 = N(m_{in}p(1-p) + p^2)$ high and $n_0 = Np - n_1$ low neurons of $\mathbf{f}^1$. Thus, for high neurons

$$\langle \Sigma_i^1 \rangle = \frac{MC}{L}[(1-q)(1-q)n_1 + (1-q)(q'-q)n_0]$$

and for low neurons

$$\langle \Sigma_i^1 \rangle = \frac{MC}{L}[(q'-q)(1-q)n_1 + (q'-q)(q'-q)n_0]$$

where $MC/L = \langle M^1 \rangle$ and

$$q' = 1 - (1-p)^{C-1} = 1 - (1-q)/(1-p)$$

is the probability of the neuron to be active due to the presence of other factors except $\mathbf{f}^1$ in the learning pattern. Therefore,

$$\langle \Sigma_i^1 \rangle = \frac{MNCp(1-q)^2}{L(1-p)}(f_i^1 - p)m_{in}.$$

Since $\mathbf{X}^{in}$ is independent of all factors except $\mathbf{f}^1$,

$$\langle \Sigma_i^0 \rangle = M(1 - C/L)(N-1)p\langle (X_i^m - q^m)(X_j^m - q^m) \rangle$$

where $M(1 - C/L) = \langle M^0 \rangle$. By the definition of q^m, for each pattern of the learning set $\sum\limits_{i=1,N}(X_i^m - q^m) = 0$. Hence

$$\langle (X_i^m - q^m)(X_j^m - q^m) \rangle = -\langle (X_i^m - q^m)^2 \rangle/(N-1) = -q(1-q)/(N-1). \tag{3.2}$$

For $N \to \infty$ and $L \to \infty$ $\langle \Sigma_i^0 \rangle = -Mpq(1-q)$ and, therefore, the mean synaptic excitations amount to

$$\langle h_i \rangle = \frac{MNCp(1-q)^2}{L(1-p)}(f_i^1 - p)m_{in} - Mpq(1-q). \tag{3.3}$$

The mean synaptic excitation for high neurons ($f_i^1 = 1$) is larger than that for low neurons ($f_i^1 = 0$). Thus the high neurons have higher probability to be active at the next step of the network dynamics.

To estimate the probabilities of high and low neurons to be active, let us now estimate the dispersions of synaptic excitations. Since $M^0 \gg M^1$, these dispersions are determined by the dispersion of Σ_i^0. Therefore,

$$D\{h_i\} = NpD\{J_{ij}\} + N^2p^2Cov\{J_{ij}, J_{ik}\}.k \neq j \neq i \tag{3.4}$$

As shown in Appendix 3

$$D\{J_{ij}\} = \frac{M^2C^2p^2(1-q)^4G(\mu)}{L(1-p)^2} \tag{3.5}$$

where $\mu = C^2/L$,

$$G(\mu) = [\exp(\mu(\frac{1}{(1-p)^2} - 1)) - 2\exp(\mu(\frac{1}{1-p} - 1)) + 1](1-p)^2/(\mu p^2) \tag{3.6}$$

and function $G(\mu)$ is chosen so that $G(0) = 1$. As shown in Fig 1a in the range $0 < \mu < 1$ this function is close to the straight line and tends to the line $G = 1 + \mu$ when sparseness increases.

To estimate $Cov\{J_{ij}, J_{ik}\}$ one can notice that according to the correlational Hebbian rule

$$\sum_{j=1,N,j\neq i} J_{ij} = -\sum_{m=1,M} (X_i^m - q^m)^2.$$

Thus

$$\begin{aligned}(N-1)D\{J_{ij}\} + (N-1)(N-2)Cov\{J_{ij}, J_{ik}\} = -MD\{(X_i^m - q^m)^2\} \\ -M(M-1)Cov\{(X_i^m - q^m)^2, (X_i^l - q^l)^2\}.\end{aligned} \tag{3.7}$$

Both terms on the right side of (3.7) that are respectively of order M and M^2, can be ignored when compared with the first term on the left side of this equation that is of order NM^2. Hence

$$Cov\{J_{ij}, J_{ik}\} = -\frac{D\{J_{ij}\}}{N} \tag{3.8}$$

and according to (3.4) and (3.5),

$$D\{h_i\} = \sigma^2 = Np(1-p)D\{J_{ij}\} = Np(1-p)\frac{M^2C^2p^2(1-q)^4G(\mu)}{L(1-p)^2} \tag{3.9}$$

The network with complex learning patterns is reduced to the ordinary Hopfield network with simple learning patterns when $C = 1$ and $M = L$. In this case $\mu = 0$, consequently $G(\mu) = 1$ and the expressions for means and variance of synaptic excitations coincide with those for sparsely encoded ordinary Hopfield network [8].

In the limit case $N \to \infty$, the distributions of synaptic excitations can be approximated by normal ones. Then at the first step of the recall process

$$Prob\{X_i(1) = 1\} = \Phi(\theta_i)$$

where

$$\theta_i = (T(1) - \langle h_i \rangle)/\sigma,$$

$$\Phi(x) = 1/(2\pi)^{1/2} \int_x^\infty \exp(-u^2/2)du \tag{3.10}$$

and $T(1)$ is an activation threshold. According to (3.3) and (3.9) for high and low neurons of $\mathbf{f}^1$

$$\begin{aligned}\theta^1 &= \theta - \frac{m_{in}(1-p)}{\sqrt{Lp(1-p)G(\mu)/N}} = \theta - \frac{m_{in}(1-p)}{\sqrt{\gamma p(1-p)/I(p)}}, \\ \theta^0 &= \theta + \frac{m_{in}p}{\sqrt{\gamma p(1-p)/I(p)}}\end{aligned}$$

where $\theta = (T(1) + Mpq(1-q))/\sigma$ is a scaled activation threshold and

$$\gamma = \alpha G(\mu). \tag{3.11}$$

In the model the threshold is chosen in such a way that a total level of the network activity is the same as in factors, i.e. is chosen to satisfy condition

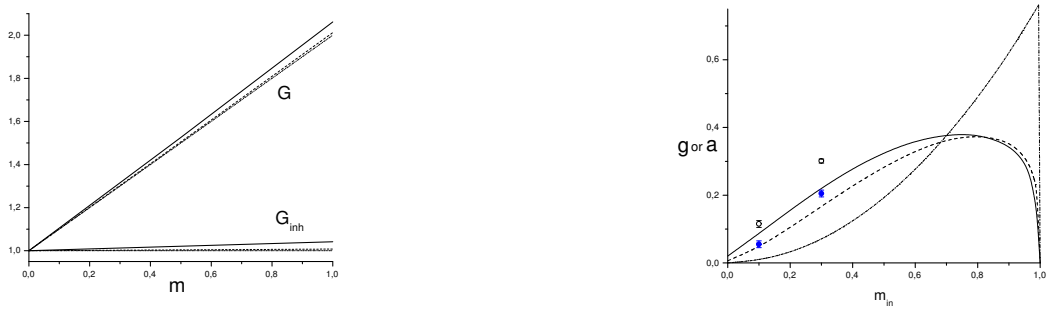
$$pp_1 + (1-p)p_0 = p \tag{3.12}$$

where $p_1 = \Phi(\theta_1)$ and $p_0 = \Phi(\theta_0)$ are probabilities for high and low neurons to be active. As a result of the first step, the overlap changes to

$$m(1) = p_1 - p_0. \tag{3.13}$$

In the single-step approximation these equations are assumed to be valid for all time steps (naturally, m_{in} must be replaced by $m(t)$ and $m(1)$ by $m(t+1)$ where t and $t+1$ are consequence steps of the recall process).

The obtained neurodynamic equations completely coincide with those for ordinary Hopfield network [8] if γ is replaced by α. Parameter γ completely determines the network dynamics in the SS approximation for given m_{in} and p. The curves which characterize the behavior of the network activity depending on γ, m_{in} and p are presented in Fig.1b. Let the initial state of the network activity for a given γ be characterized by the point (m_{in}, γ). If this point lays under the curve, the overlap between the current pattern and the recalled pattern sways during the recall process to the right, that is to the final overlap m_f given by the right branch of the curve. The overlap sways to the left for each point above the curve. Thus the left branch of the curve corresponds to the border of an attraction basin. In the SS approximation this border corresponds to the condition $m(1) = m_{in}$.



(a) **(b)**

Fig. 1 *Results of SS approximation (lines) and computer simulation (points).* $p = 0.02$ *- solid line,* $p = 0.004$ *- dashed line,* $p \to 0$ *- dashed-dotted line;* $\circ$ *-* $p = 0.02$, $\bullet$ *-* $p = 0.004$. *a) Functions* $G(\mu)$ *and* $G_{inh}(\mu)$ *taking account of signals complexity without and with additional inhibition. b) Sizes of attraction basins in dependence on relative informational loading* α *and signals complexity* μ, $\gamma = \alpha G(\mu)$ *is a combined index of their influence to attraction basins if additional inhibition is not used. If additional inhibition is used,* $\gamma = \alpha$.

Fig. 1b demonstrates that in the SS approximation for given γ the size of the attraction basins monotonically decreases when the encoding sparseness increases. Computer simulations showed that equations (3.3) and (3.4) provide quite accurate estimations for the means and variances of synaptic excitations when N exceeds 10^3 and thus these equations quite accurately predict the first step of neurodynamics.

4 Multi-step retrieval

Properties of the multi-step retrieval were investigated by computer simulation that was performed for N from 1100 to 10000. The program generated a set of random factors of fixed sparseness p and mixed them into a set of M patterns so that each pattern contained exactly the C factors. The network was trained by this set and then tested by corrupted versions of factors for up to 20000 simulation trials for each combination of parameters. The most results were obtained for $p = 0.02$ and $m_{in} = 0.3$.

The sizes of attraction basins appeared to be very far from SS prediction. Fig. 2a illustrates trajectories of neurodynamics for $M = 40000$, $N = 1100$, $\alpha = 0.1$, $C = 20$ on the plane constituted by axes $[m(t), \Lambda(t)]$ where $m(t)$ is the overlap of the network state with the recalled factor and $\Lambda(t)$ is the Lyapunov function calculated by (1.5). In accordance with the SS prediction the most trajectories displaced at the first step to the recalled factors. Averaged over all trajectories $m(1) = 0.368 \pm 0.003$ while SS gives $m(1) = 0.41$. Discrepancy between the experimental results and the SS prediction decreases when N increases. However in contrast to the main SS assumption $m(t)$ does not change monotonically and at the next steps most trajectories returned back and ended far from the recalled factors. This neurodynamics is similar to the observed one for the densely encoded ordinary Hopfield network (see, for example,[1]). Note that the Lyapunov function of spurious attractors is much larger

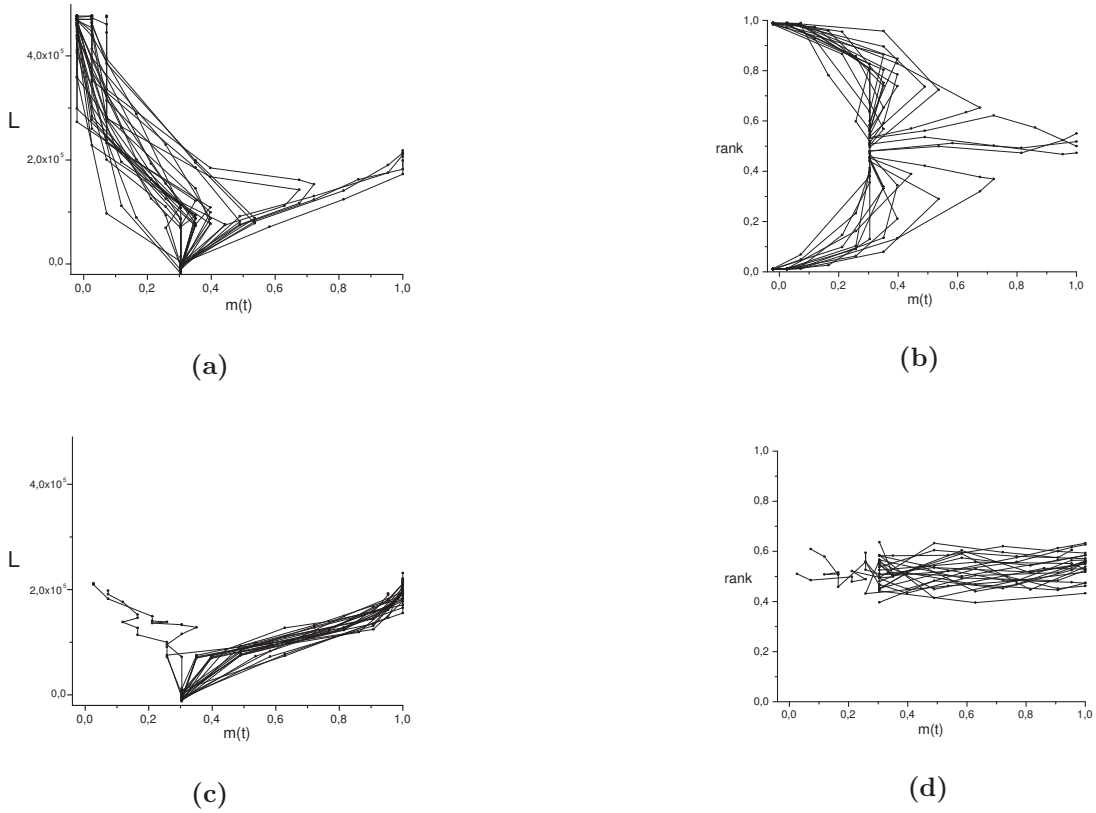


Fig. 2 *Trajectories of neurodynamics without (a) and (b) and with (c) and (d) additional inhibition obtained for* $M = 40000$, $N = 1100$, $\alpha = 0.1$, $C = 20$ *and* $m_{in} = 0.3$. *Abscissa is overlap* $m(t)$ *between the current network activity and the recalled factor. Ordinate is Lyapunov function (a) and (c) or a rank of the network activity (b) and (d) showing contribution of neurons most often or rarely contained in factors set. Additional inhibition completely suppressed the dominance of two global attractors.*

than that of true attractors.

The spurious attractors are two global attractors created by neurons contained in the most and the least numbers of factors, respectively. To demonstrate this fact we redrew the graph in the axes $[m(t), c(t)]$ where $c(t)$ indicates whether the neurons contained in the most or least numbers of factors contribute to the current network activity. To calculate $c(t)$, we ranged all neurons in the order of numbers of factors that contained them. So the neurons that were contained in the least number of factors had the least rank and contained in the most number of factors had the highest rank. The rank of the current activity was calculated as a sum of ranks of active neurons. The obtained rank was normalized so that the patterns created by the neurons contained in the least number of factors have rank c close to zero, created by the neurons contained in the most number of factors have $c \simeq 1$ and created by random neurons have $c \simeq 0.5$. Fig. 2b demonstrates that the patterns created by the global spurious attractors had ranks c close to 0 and 1, while true attractors had ranks around 0.5.

4.1 Lyapunov function of true and global spurious attractors

Two global spurious attractors dominate because their Lyapunov function exceeds the one of true attractors. By definition, the Lyapunov function of each attractor can be estimated as $\Lambda = nh$ where $n = pN$ is a number of active neurons and h is mean synaptic excitation produced in these neurons by their proper activity. Since true attractors almost coincide with factors, their mean synaptic excitation

can be estimated by (3.3) for $f_i = 1$ and $m_{in} = 1$. Thus the Lyapunov function of true attractors is

$$\Lambda_{tr} = M[Np(1-q)]^2 C/L - MNp^2 q(1-q) \tag{4.1}$$

Let us estimate now the Lyapunov function of global spurious attractors. Let k be the number of factors containing a given neuron. The mean and variance of k are pL and $p(1-p)L$, respectively. Then the number of neurons which belong to $k > k_1$ factors can be estimated as $N\Phi(u_1)$ where $u_1 = (k_1 - pL)/\sqrt{p(1-p)L}$. To choose pN neurons with the largest k from totally N neurons, one must put $k_1 = pL + u_1\sqrt{p(1-p)L}$ where u_1 satisfies equation $\Phi(u_1) = p$. On average, each of the chosen neurons belongs to $k_2 = pL + u_2\sqrt{p(1-p)L}$ patterns where

$$u_2 = \frac{1}{\Phi(u_1)\sqrt{2\pi}} \int_{u_1}^{\infty} u \exp(-u^2/2) du.$$

The probability of one of these neurons to be active during the presentation of a learning pattern is $r \simeq 1 - \exp(-k_2 C/L) \simeq q + (1-q) C u_2 \sqrt{p(1-p)/L}$. Then a mean augmentation of synaptic connection between two of these neurons during the presentation of input pattern is $\Delta J = (r-q)^2$ and a mean strength of connection after presentation of the whole learning set is $J = M\Delta J = M(r-q)^2$. Hence the Lyapunov function for this attractor can be estimated as

$$\Lambda_{sp}^{gl} \simeq M(pN)^2 J = M[Np(r-q)]^2 \simeq M[Np(1-q)Cu_2]^2 p(1-p)/L.$$

Since $p \ll 1$ then $u_2 \simeq u_1 \simeq [-2\ln(p\sqrt{2\pi})]^{\frac{1}{2}}$. Consequently

$$\Lambda_{sp}^{gl} \simeq 2M[Np(1-q)C]^2 p \ln(1/[p\sqrt{2\pi}])/L \tag{4.2}$$

Similarly, it is easy to estimate Lyapunov function for the attractor created by neurons belonging to the smallest number of factors. To do this it is necessary to replace k_2 in the formula for r by $k_3 = pL - u_2\sqrt{p(1-p)L}$ keeping all other equations. This results in the same expression for Lyapunov function as (4.2).

According to (4.1) and (4.2) Lyapunov function of true attractors increases proportionally to C while of spurious attractors proportionally to C^2. That is why spurious attractors dominate for large C and are not observed for small C, particularly for ordinary Hopfield network with $C = 1$.

Usually second term in (4.1) is relatively small and can be ignored. Then

$$\Lambda_{sp}^{gl}/\Lambda_{tr} \simeq 2Cp \ln(1/[p\sqrt{2\pi}]) \tag{4.3}$$

Fig. 3 demonstrates the ratio of Lyapunov functions for spurious and true attractors in dependence on C and p obtained by (4.3) and their experimental ratios obtained for $p = 0.02$ and different N and α. It is shown, first that (4.3) gives a rather accurate estimation of this ratio, second that the ratio only slightly depends on N and α, and third that the critical complexity, when spurious attractors become dominant, amounts to about $C = 10$ for $p = 0.02$ and increases when sparseness increases. According to (4.3), when $p \to 0$, both effects of the complexity and sparseness on global spurious attractors can be taken into account by single parameter $Cp\ln(1/p)$. Since in this case Shannon function $H(p) \simeq p\ln(1/p)$, the use of this parameter is equivalent to the use of parameter $CH(p)$, i.e. for very large sparseness it is reasonable to normalize complexity by the Shannon function as was done before for informational loading.

4.2 Suppression of global attractors

In order to provide the factors retrieval for large complexity, two global spurious attractors have to be suppressed. Since the strength of connection between the neurons of these attractors is given by $M(r-q)^2$ where r is a probability that each of these neurons is active in the learning pattern, the suppression of these attractors can be achieved by substraction of matrix $\mathbf{J}'$ from the connection matrix $\mathbf{J}$ obtained by Hebbian rule (2.1) where

$$J'_{ij} = M(r_i - q)(r_j - q), i \neq j, \qquad J'_{ii} = 0 \tag{4.4}$$

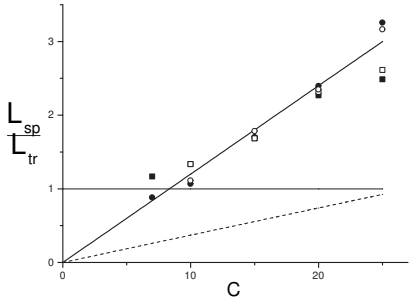


Fig. 3 *The ratio of values of Lyapunov function for global spurious and true attractors. Estimations by (4.3) are shown by a thick solid line for* $p = 0.02$ *and a dashed line for* $p = 0.004$*. Points are experimental data averaged over four random sets of factors for* $M = 40000$*,* $N = 1100$*; squares -* $\alpha = 0.1$*, circles -* $\alpha = 0.2$*, open and full points - attractors created by neurons the most rare and often contained in factors, respectively. The thin horizontal line indicates the equality of values of Lyapunov function for spurious and true attractors. For* $p = 0.02$ *this line is crossed for* $C \approx 10$ *indicating the critical value of complexity when spurious attractors become dominate.*

and r_i is a frequency of appearance of the i-th neuron in the learning set. Then connections between neurons that were extremely often or rarely presented in the learning set are selectively reduced and do not change for the most neurons whose activity was close to the mean level q.

In the frame of neural network approach, substraction of $\mathbf{J}'$ can be implemented by additional inhibitory neuron bi-directionally connected with all principal neurons of the network by connection vector $\bar{\mathbf{J}}$. In the learning stage this neuron is activated at each presentation of the learning pattern and its connections are modified by the Hebbian rule. Then, as a result of storing all M patterns of the learning set,

$$\bar{J}_i = \sum_{m=1}^{M} (X_i^m - q^m) = M(r_i - q).$$

Let us also assume that the excitability of the inhibitory neuron decreases during the learning as to $1/M$. In the recall stage its activity is then

$$A(t) = (1/M) \sum_{i=1}^{N} \bar{J}_i X_i(t) = (1/M) \bar{\mathbf{J}}^T \mathbf{X}(t)$$

where $\bar{\mathbf{J}}^T$ is transposed $\bar{\mathbf{J}}$. Respectively, the inhibition produced in all principal neurons of the network is given by vector $\bar{\mathbf{J}} A(t) = (1/M)\bar{\mathbf{J}}\bar{\mathbf{J}}^T\mathbf{X}(t)$. Since $\mathbf{J}' \simeq (1/M)\bar{\mathbf{J}}\bar{\mathbf{J}}^T$ (the difference only in diagonal elements), this inhibition is equivalent to the substraction of $\mathbf{J}'$ from $\mathbf{J}$.

Fig. 2c and 2d demonstrate how the trajectories of network activity change due to this inhibition. It is shown that now most trajectories converge to factors, two global spurious attractors are completely suppressed and trajectories that converge to states far from the factors are attracted by local attractors with rank $c \simeq 0.5$ and the Lyapunov function close to that for true attractors.

Another effect of the inhibition shown in Fig. 2 is improvement of convergence to factors even at the first step of neurodynamics. The displacement of trajectories to recalled factors became larger at this step than without inhibition. Averaged over all trajectories $m(1) = 0.45 \pm 0.002$ (without inhibition it was 0.38 ± 0.003). To evaluate this effect we recalculated the means and variance of synaptic excitations of principal neurons with additional inhibition. As shown in the previous section they are completely determined by the mean and variance of synaptic connections. Thus, to estimate how inhibition modifies the SS prediction it is enough to estimate how it modifies connection matrix.

According to (4.4), $\langle J'_{ij} \rangle = 0$. Thus inhibition does not modify the mean of synaptic connections J_{ij}. However, as shown in Appendix 4, it significantly reduces their variance. Particularly the variance

of $J_{ij}^{inh} = J_{ij} - J'_{ij}$ amounts to:

$$D\{J_{ij}^{inh}\} = \frac{M^2C^2p^2(1-q)^4G_{inh}(\mu)}{L(1-p)^2} \tag{4.5}$$

where

$$G_{inh}(\mu) = [\exp(\mu(\frac{1}{(1-p)^2} - 1)) - \exp(2\mu(\frac{1}{1-p} - 1))](1-p)^2/(\mu p^2). \tag{4.6}$$

Thus the variances of effective synaptic connections with (J_{ij}^{inh}) and without (J_{ij}) inhibition differ only by functions $G_{inh}(\mu)$ and $G(\mu)$ given by (4.6) and (3.6), respectively. Function $G_{inh}(\mu)$ is compared with $G(\mu)$ in Fig. 1a. It is shown that due to the inhibition the variance of effective synaptic connections becomes only slightly dependent on μ and G_{inh} tends to line $G_{inh}(\mu) = 1$ when sparseness increases. Thus, inhibition actually improved parameters of the SS approximation making them close to those of the ordinary Hopfield network with $C = 1$ and therefore improves the network dynamics even at the first step of neurodynamics.

4.3 Size of attraction basins

Fig. 2 also demonstrates that as for the ordinary Hopfield network, the borders of the attraction basins around the factors are fuzzy: starting from the states with the same m_{in}, the trajectories may converge to the recalled factor or to some spurious state far from all factors. Consequently, the distribution of final overlaps has two distinct modes: $m_f \approx 1$ ("true") and $m_f << 1$ ("spurious"). It is well known for the ordinary Hopfield network that for small informational loading a "true" mode prevails, and as informational loading increases, the distribution maximum shifts into a "false" mode, demonstrating a sharp transition from a retrieval to a not-retrieval network dynamics at a certain $\alpha = \alpha_{ab}$. The transition becomes more sharp when network size increases.

To estimate the critical informational loading α_{ab} for given m_{in} we used the same method as for the ordinary Hopfield network [9]. Particularly, we first estimated probabilities of correct recall P in dependence on the size of a learning set M, informational loading α and network size N for fixed C and m_{in}. Then we approximated this dependence by a special regression model and extrapolated it for $N \to \infty$ to find critical α_{ab}.

For each set of parameters the probability was estimated as a portion of true trajectories at the histogram of m_f distribution. In order to separate "true" and "spurious" modes at the histogram, we used the border $m_f = 0.72$. Since the "true" and "spurious" modes were always well separated, an exact choice of the border was not important. The computed values of P were transformed by the following logistic mapping to variable F:

$$P = \frac{1}{1 + e^{-F}} \tag{4.7}$$

The advantage of this transformation is that F has no limits, whereas $0 \le P \le 1$. Thus, to approximate $F(\alpha, N, M)$ for given C and α by some regression model it is not required to use any constrains to restrict F as it would be required for direct approximation of $P(\alpha, N, M)$.

As an example, the dependence of F on M is shown in Fig. 4 for $N = 3000$, $C = 20$ and $m_{in} = 0.3$. Data were obtained with a connection matrix modified by additional inhibition. As shown in Fig.4, F can be well fitted by linear dependence on $1/M$. Intercepts of the regression lines that approximate the dependence of F on $1/M$ were used to estimate the asymptotic values of F for $M \to \infty$.

As was shown previously, the variance of synaptic connections lost dependence on signal complexity due to additional inhibition. Since the network behavior depends only on the effective connection matrix, one could suggest that its behavior had lost dependence on signal complexity in general. This suggestion is confirmed by the results of factors recalling shown in Fig. 5. This figure presents dependence of F on N and α for $m_{in} = 0.3$ and $C = 20$ and 1. For $C = 20$, only asymptotic values of F for $M \to \infty$ obtained with inhibition are shown. For $C = 1$ the data are shown obtained with and without inhibition. Thereby for $C = 1$ the recall procedure was performed with matrices $\mathbf{J}$ and

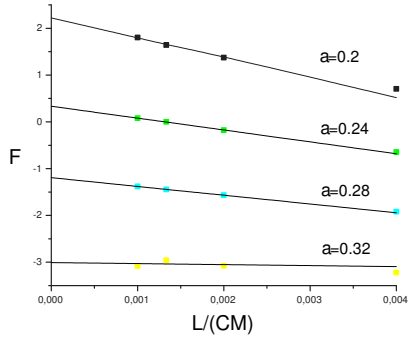


Fig. 4 *Dependence of transformed probability F of trajectories convergence to factors in dependence on the size of learning set M for $N = 3000$, $C = 20$ and $m_{in} = 0.3$. Intercepts with axes of the ordinate were used as experimental estimations of transformed probability for $M \to \infty$.*

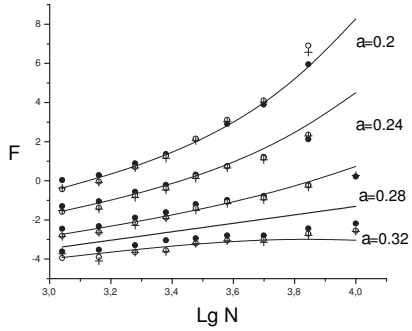


Fig. 5 *Comparison of transformed probabilities F of trajectories convergence to factors in dependence on network size N and informational loading α for $C = 20$ with additional inhibition (•) and for $C = 1$ with (○) and without (+) additional inhibition, $m_{in} = 0.3$. Thin lines correspond to regression model (4.8). The thick straight line corresponds to the critical value of informational loading $\alpha_{ab} = 0.303$.*

$\mathbf{J}^{inh} = \mathbf{J} - \mathbf{J}'$ where $\mathbf{J}$ is an ordinary Hopfield connection matrix and $\mathbf{J}'$ is given by (4.4). As shown in Fig. 5, the data actually form a homogeneous family, although the ability to recall stored factors happened to be even higher for $C = 20$ than for ordinary Hopfield network (it becomes especially remarkable when informational loading increases). One of the possible explanations is that storing of complex patterns produces noise in the connection matrix and this noise suppresses some local spurious attractors of the ordinary Hopfield network. Note also that the additional inhibition does not influence the dynamic properties of the ordinary Hopfield network.

Since the data were close for all three groups ($C = 20$ with inhibition and $C = 1$ with and without inhibition) they were combined in one family of data for approximation by the regression model

$$F = a_0 + a_1\alpha + a_2 N + a_3 \ln N + a_4 \alpha N. \tag{4.8}$$

As shown in [9] this regression model allows for rather accurate extrapolation of dependence $F(\alpha, N)$ obtained by the fit of data for relatively small network size $N \leq 10^4$ to the range of a very large network size up to $N = 10^5$. The fitted curves that approximate the data for fixed α are also shown in Fig. 5 by thin lines. According to this approximation, the lines constitute two groups. The upper lines are concave and tend to $+\infty$ when $N \to \infty$. The lower line is convex and tends to $-\infty$ when $N \to \infty$. Transition from one to another group occurs due to the change of α. The value of α which corresponds to the thick straight line separating these groups is chosen as critical α_{ab}. For

each $\alpha < \alpha_{ab}$ the probability of trajectories to converge to factors tends to 1 when N increases. And conversely for $\alpha > \alpha_{ab}$ it tends to zero. Thus α_{ab} corresponds to sharp transition from retrieval to nonretrieval conditions for $N \to \infty$.

¿From the regression model, α_{ab} can be evidently found as $\alpha_{ab} = -a_2/a_4$. For data combined in a joint family $\alpha_{ab} = 0.307 \pm 0.006$. This value is shown in Fig. 1b. The regression model, applied to each of three groups of data separately, gives $\alpha_{ab} = 0.303 \pm 0.005$ for $C = 20$, $\alpha_{ab} = 0.315 \pm 0.008$ for the ordinary Hopfield network with additional inhibition and $\alpha_{ab} = 0.307 \pm 0.008$ for the ordinary Hopfield network. All these values differ nonsignificantly. However, they significantly exceed the value $\alpha_{ab} = 0.22$ predicted by SS (see Fig. 1b). Thus for sparse encoding ($p = 0.02$) the SS approximation underestimates the size of attraction basins. This is confirmed by computer simulation performed for $p = 0.02$ and $m_{in} = 0.1$ and for $p = 0.004$, $m_{in} = 0.1$ and 0.3 for the ordinary Hopfield network. The obtained estimations are also shown in Fig. 1b. When α is smaller than α_{ab} predicted by SS, then $m(t)$ increases monotonically according to SS assumption. But when α is larger than α_{ab} predicted by SS but smaller than α_{ab} obtained experimentally, then $m(t)$ changes nonmonotonically: trajectories move away from the recalled factors at the first step but then return and end in their vicinities. Without additional inhibition they usually demonstrate opposite behavior: $m(t)$ increases at the first step but then decreases.

5 Discussion

Theoretical analysis and computer simulations revealed that the Hopfield-like neural networks are capable of performing Boolean factor analysis of the signal of a high dimension and complexity. This ability is based on the fact that, due to the correlational Hebbian rule, the factors become attractors of the network dynamics. This is identical with the property of the factors to be eigenvectors of the correlational matrix in linear factor analysis. However, in contrast to the linear factor analysis, the number of binary factors can be much larger than the patterns dimensionality.

Both the SS approximation and the computer simulation revealed that this capability is mainly determined by two network parameters: relative informational loading α and complexity of a learning set C. The ability worsens when both these parameters increase. However, the computer simulation has shown that the SS predictions are very far from reality. The network loses this ability when α and C are much smaller then the SS predicts. This failure of the SS approximation is explained by the existence of two global spurious attractors that SS fails to predict. These attractors become to dominate when signal complexity is large independently of informational loading. For example, for $p = 0.02$ the critical complexity of their dominance amounts to $C \simeq 10$. The critical complexity increases when sparseness increases.

The dominance of the global attractors can be completely suppressed by addition to the network of a single inhibitory neuron with bi-directional Hebbian synapses. Due to this additional inhibition, the network completely loses its dependence on signal complexity and all properties of network dynamics becomes identical to those for the ordinary Hopfield network. The size of the attraction basins happened to be much closer to the SS prediction but contrary to the case without inhibition it exceeds the SS prediction now. If the initial network state is inside the attraction basin obtained by computer simulation but outside predicted by SS, then at the first step of the neurodynamics, trajectory goes out of the recalled factor but then turns back and tends to it. Such behavior is typical the for sparsely encoded ordinary Hopfield network [9] and opposite to the observed one for the densely encoded Hopfield network [1]. For the densely encoded network the SS prediction can be significantly improved by the method of Statistical Neurodynamics (SN) elaborated by Amari and Maginu [1]. However, for the sparsely encoded network the prediction of SN is even worse than the SS. Thus until now, SS is the most accurate method to analyze sparsely encoded Hopfield-like networks. Computer simulation has shown that its accuracy increases when sparseness increases (compare the data for $p = 0.02$ and $p = 0.004$ in Fig. 1b).

On the whole, our results suggest that for a sufficiently large range of parameters the neural network approach seems to be prospective for developing a general statistical method for a binary factor analysis. The present approach is in line with many recent attempts to elaborate new statistical methods for nonlinear factor analysis; for example, nonlinear Independent Component Analysis ([13]).

Some of these methods utilize neural networks [12] to perform iterative algorithms; however, the learning rules and network dynamics employed are mostly artificially made up to provide the required computations. A major distinction of the current work from the related ones is that natural properties of unmodified Hebbian learning and dynamics of attractors neural network are implemented in a simple paradigm which has clear neurobiological interpretation.

There are many examples of data in the sciences when Boolean factor analysis is required [7]. However, binary factor analysis with the use of neural network approach seems especially efficient for processing textual data. Since the time of the factor search only slightly increases with the increase of signal space dimensionality, it is most attractive to apply this approach to patterns of very large dimensionality. Texts are good examples of such a kind of signals. The dimensionality of signal space is equal to the number of words in the used dictionary and thus is usually very high.

This paper only emphasizes the principal ability of the Hopfield-like network to perform Boolean factor analysis. This ability is based on the fact that a network keeps its ability to create large attraction basins around the stored patterns even in the case when they are mixed in the signals of a learning set. The realistic procedure of Boolean factor analysis with the Hopfield-like network is presented in the accompanying paper.

6 Appendix 1. Attractors of network dynamics

In order to prove that only point and cyclic attractors of length two are present in the network dynamics, let us introduce the function

$$F(t) = \Lambda(t) - (\mathbf{X}^T(t+1) + \mathbf{X}^T(t))\mathbf{T}^n$$

where the first term is an ordinary Lyapunov function $\Lambda(t) = \mathbf{X}^T(t+1)\mathbf{J}\mathbf{X}(t)$ and the second term takes account of small noisy increments T_i^n added to the global activation threshold $T(t)$. These increments are so chosen as to be different for different neurons and fixed during the whole recall process. Since $\mathbf{J}$ is symmetric, $\mathbf{X}^T(t+1)\mathbf{J}\mathbf{X}(t) = \mathbf{X}^T(t)\mathbf{J}\mathbf{X}(t+1)$. Then the increment of F during one recall step amounts to

$$\Delta \quad = \quad F(t+1) - F(t) = (\mathbf{X}^T(t+2) - \mathbf{X}^T(t))(\mathbf{J}\mathbf{X}(t+1) - \mathbf{T}^n)$$

Since the number of active neurons is set to be fixed at each time step of the recall process, then $(\mathbf{X}^T(t+2) - \mathbf{X}^T(t))\mathbf{e} = 0$ where $\mathbf{e}$ is the vector of the N ones. Hence

$$\Delta = (\mathbf{X}^T(t+2) - \mathbf{X}^T(t))(\mathbf{J}\mathbf{X}(t+1) - \mathbf{T}^n - T(t+1)\mathbf{e}) = \sum_i \delta_i$$

where $\delta_i = (X_i(t+2) - X_i(t))u_i(t+1)$ and $u_i(t) = \sum_j J_{ij}X_j(t) - T_i^n - T(t)$. Individual increments δ_i are non-negative because if $u_i(t+1) > 0$, then, according to (2.2), $X_i(t+2) = 1$ and $X_i(t+2) - X_i(t) \geq 0$ regardless, of $X_i(t)$; if $u_i(t+1) < 0$ then, according to (2.2), $X_i(t+2) = 0$ and $X_i(t+2) - X_i(t) \leq 0$, regardless of $X_i(t)$. Thus in both cases $\delta_i \geq 0$ and therefore $\Delta \geq 0$. Since the number of the network states is finite, Δ finally reaches the zero value when all $\delta_i = 0$. Due to the small random noise the global activation threshold can be chosen at each time step so that $u_i \neq 0$ for all i. Then the equality $\delta_i = 0$ can be satisfied only if $X_i(t+2) = X_i(t)$. Thus the network dynamics finally reaches the point or cyclic attractors of length two.

7 Appendix 2. Estimation of $D\{q^m\}$

By definition, $q^m = (1/N)(\sum_{i=1,N} X_i^m)$. Then

$$D\{q^m\} = \frac{1}{N} D\{X_i^m\} + \frac{N-1}{N} Cov\{X_i^m, X_j^m\}$$

where $D\{X_i^m\} = q(1-q)$. Covariation between X_i^m and X_j^m can be presented in the form $Cov\{X_i^m, X_j^m\} = \langle(1-X_i^m)(1-X_j^m)\rangle - (1-q)^2$. Since factors are statistically independent

$$\langle(1-X_i^m)(1-X_j^m)\rangle = \langle \prod_{l\in\{l:\beta_l^m=1\}} (1-f_i^l)(1-f_j^l)\rangle = [(1-p)(1-p^*)]^C$$

where $1-p^* = ((1-p)N-1)/(N-1)$ is a probability that the j-th neuron is not active in a given factor under the condition that the i-th neuron is not active in this factor. Thus

$$Cov\{X_i^m, X_j^m\} = (1-q)^2[(1-\frac{p}{(1-p)(N-1)})^C - 1] \simeq -\frac{(1-q)^2pC}{(1-p)N},$$

that is $D\{q^m\}$ is of order $1/N$.

8 Appendix 3. Estimation of $D\{J_{ij}\}$

$D\{J_{ij}\}$ can be presented in the form

$$D\{J_{ij}\} = ME_1 + M(M-1)E_2 - (\langle J_{ij}\rangle)^2 \tag{8.1}$$

where

$$\begin{aligned} E_1 &= \langle(X_i^m - q^m)^2(X_j^m - q^m)^2\rangle, \\ E_2 &= \langle(X_i^m - q^m)(X_j^m - q^m)(X_i^l - q^l)(X_j^l - q^l)\rangle, \qquad l \neq m \end{aligned}$$

and according to (3.2) $\langle J_{ij}\rangle = M\langle(X_i^m - q^m)(X_j^m - q^m)\rangle = -Mq(1-q)/(N-1)$.

To estimate E_1 and E_2 we ignore the statistical dependence between X_i^m and X_j^m which results from the fact that the number of active neurons in factors is fixed and equal to n (the correlation coefficient between these variables is of order $1/N$, see Appendix 2). Then $E_1 = q^2(1-q)^2$. To estimate E_2 one must take into account the statistical dependence between the activities of the same neurons in different patterns of the learning set. This dependence results from the fact that the different neurons are differently presented in a set of factors. Thus the neurons which are contained in more factors have higher probability to be active in both learning patterns $\mathbf{X}^m$ and $\mathbf{X}^l$. In order to take into account this dependence explicitly, let us introduce probability

$$P(C_1) = \binom{C}{C_1}\binom{L-C}{C-C_1}/\binom{L}{C}$$

that the given pair of signals have C_1 common factors.Then

$$E_2 = \sum_{C_1} P(C_1)E^2(C_1)$$

where

$$E(C_1) = \langle(X_i^m - q^m)(X_i^l - q^l)\rangle|_{C_1} = \langle(1-X_i^m)(1-X_i^l)\rangle - (1-q)^2.$$

Due to independence of different factors

$$\langle(1-X_i^m)(1-X_i^l)\rangle = (1-p)^{C_1}(1-p)^{2(C-C_1)} = (1-q)^2/(1-p)^{C_1},$$

i.e. $E(C_1) = (1-q)^2[(1-p)^{-C_1} - 1]$ and after approximation of $P(C_1)$ by Poisson distribution $P(C_1) \simeq \mu^{C_1}\exp(-\mu)/C_1!$ where $\mu = C^2/L$, and taking in account that for any a

$$\sum_{C_1} a^{C_1}\mu^{C_1}\exp(-\mu)/C_1! = \exp(\mu(a-1))$$

one can immediately obtain

$$\begin{aligned} E_2 &= (1-q)^4[\exp(\mu(\frac{1}{(1-p)^2}-1)) - 2\exp(\mu(\frac{1}{1-p}-1)) + 1] \\ &= (1-q)^4 p^2 G(\mu)\mu/[(1-p)^2] \end{aligned}$$

where $G(\mu)$ is given by (3.6). Since the first term in (8.1) is of order M, the second one is of order M^2 and the third one is of order $(M/N)^2$, the first and third terms can be ignored when compared with the second one. Hence $D\{J_{ij}\}$ is given by (3.5).

It must be noted that the estimation of $D\{J_{ij}\}$ by formula (3.5) is valid only when $1-q$ is not extremely small, because the first term in (8.1) is of order $(1-q)^2$, the second of $(1-q)^4$ and the third of $(1-q)^2$. Since $1-q = \exp(-pC)$, we assume that pC is not extremely large to provide the condition that $(1-q)^2$ and $(1-q)^4$ are of the same order.

9 Appendix 4. Estimation of mean and variance of synaptic connections with additional inhibition

The estimations are performed by the same method as in Appendix 3. According to (2.1) and (4.4)

$$\begin{aligned} J_{ij}^{inh} &= J_{ij} - J'_{ij} = \sum_{m=1}^{M}(X_i^m - q^m)(X_j^m - q^m) - M(r_i - q)(r_j - q) \\ &\simeq \sum_{m=1}^{M}(X_i^m - q)(X_j^m - r_j) = \sum_{m=1}^{M}(1-X_i^m)((1-X_j^m) - (1-r_j)) \end{aligned}$$

where r_i is the probability that neuron i is active in a learning pattern. Then $D\{J_{ij}^{inh}\} \simeq M^2 E_2^{inh}$ where

$$\begin{aligned} E_2^{inh} &= \langle(1-X_i^m)((1-X_j^m)-(1-r_j))(1-X_i^l)((1-X_j^l)-(1-r_j))\rangle \\ &= \langle(1-X_i^m)(1-X_j^m)(1-X_i^l)(1-X_j^l)\rangle - \langle(1-X_i^m)(1-r_j)(1-X_i^l)(1-X_j^l)\rangle \\ &- \langle(1-X_i^m)(1-X_j^m)(1-X_i^l)(1-r_j)\rangle + \langle(1-X_i^m)(1-r_j)(1-X_i^l)(1-r_j)\rangle \\ &\simeq \langle(1-X_i^m)(1-X_j^m)(1-X_i^l)(1-X_j^l)\rangle - [\langle(1-X_i^m)(1-X_i^l)\rangle]^2 \end{aligned}$$

where we took into account that $\langle(1-X_i^m)(1-r_i)\rangle \simeq \langle(1-X_i^m)(1-X_i^l)\rangle$ and so on. The terms in last equation was found in Appendix 3 and substitution of them immediately gives (4.5).

Bibliography

[1] Amari S, Maginu K. 1988. Statistical neurodynamics of associative memory. Neural Networks, **1**:63-73.

[2] Amit DJ, Gutfreund H, Sompolinsky H. 1987. Statistical mechanics of neural networks near saturation. Annal of Physics, **173**:30-67.

[3] Bartholomew D.J, Steele F, Moustkaki I, Galbrait J. I. 2002. The Analysis and Interpretation of Multivariate Data for Social Scientists 263p.

[4] Bucingham J, Willshaw D. 1993. On setting unit thresholds in an incompletely connected associative net. Network, **4**:441-459.

[5] Buhmann J, Divko R, Schulten K. 1989. Associative memory with high information content. Physical Review A, **39**:2689-2692.

[6] De Boeck P., Rosenberg S., 1988. Hierarchical classes: model and data analysis. Psychometrika **53**:361381.

[7] De Leeuw J. 2003. Principal component analysis of binary data. Application to roll-call analysis. http://gifi.stat.ucla.edu

[8] Frolov AA, Husek D, Muraviev IP. 1997. Informational capacity and recall quality in sparsely encoded Hopfield-like neural network: Analytical approaches and computer simulation. Neural Networks, **10**:845-855.

[9] Frolov AA, Husek D, Muraviev IP. 2003. Informational efficiency of sparsely encoded Hopfield-like autoassociative memory. Optical Memory & Neural Networks, **12**,3:177-197.

[10] Goles-Chacc E, Fogelman-Soulie F, Pellegrin D. 1985. Decreasing energy functions as a tool for studying threshold networks. Discrete mathematics, **12**:261-277.

[11] Hopfield JJ. 1982. Neural network and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Science USA, **79**:2544-2548.

[12] Karhunen J, Joutsensalo J. 1994. Representation and separation of signals using nonlinear PCA type learning. Neural Networks, **7**:113-127.

[13] Karhunen J. 2001. Nonlinear Independent Component Analysis. In "Independent Component Analysis: Principles and Practice" (Eds. S. Roberts & R. Everson). Campridge University Press. p113-134.

[14] Kinzel W. 1985. Learning and pattern recognition in spin glass models. Z Physik B, **60**:205-213.

[15] Kohring GA. 1990a. A high-precision study of the Hopfield model in the phase of broken replica symmetry. Journal of Statistical Physics, **59**:1077-1086.

[16] Mickey M.R, Mundle P, Engelman, L. 1983. Boolean factor analysis. In: Dixon, W.J. (Ed.), BMDP Statistical Software. University of California Press, Berkeley, CA, p538545.

[17] Okada M. 1996. Notions of associative memory and sparse coding. Neural Networks, **9**, 98:1429-1458.

[18] Oja E, Ogawa H, Wangviwattana J. 1991. Learning in nonlinear constrained Hebbian network. In. Proc. ICANN-91, Espoo, Finland. p385-390.

[19] Watkins DS. 2002. Fundamentals of matrix computations (Second edition). John Wiley & Sons, Inc., N.Y. 411p

Contents