

Hechth–Nielsen theorem for a modified neural network with diagonal synaptic connections

Peleshchak R.¹, Lytvyn V.², Peleshchak I.², Doroshenko M.¹, Olyvko R.²

¹*Ivan Franko Drogoibych State Pedagogical University,
24 I. Franko Str., 82100, Drogoibych, Ukraine*

²*Lviv Polytechnic National University,
12 S. Bandera Str., 79013, Lviv, Ukraine*

(Received 24 September 2018; Revised 31 December 2018; Accepted 31 December 2018)

The work suggests a modified three-layer neural network with architecture that has only the diagonal synaptic connections between neurons; as a result we obtain the transformed Hecht–Nielsen theorem. This architecture of a three-layer neural network ($m = 2n + 1$ is the number of neurons in the hidden layer of the neural network, n is the number of input signals) allows us to approximate the function of n variables, with the given accuracy $\varepsilon > 0$, using one aggregation operation, whereas a three-layer neural network that has both diagonal and non-diagonal synaptic connections between neurons approximates the function of n variables by means of two aggregation operations. In addition, the matrix diagonalization of the synaptic connections leads to a decrease of computing resources and reduces the time of adjustment of the weight coefficients during the training of a neural network.

Keywords: *neural network, diagonalize the matrix, aggregation operation, approximation of function.*

2000 MSC: 68T01, 68T05, 68T10, 68T15

UDC: 004.81, 004.891.2, 004.891.3

DOI: 10.23939/mmc2019.01.101

1. Introduction

In 1900, D. Hilbert formulated a list of tasks, one of which was the problem of representing the function of n variables in the form of a superposition of functions of a smaller number of variables. In 1956, A. N. Kolmogorov [1, 2] showed that the continuous function of n variables can be represented as a superposition of functions of one variable. Later, Vitushkin [3] showed that there is no such place in the class of differentiated functions.

This question was also considered for discrete functions. In 1939, J. Slupecki [4] obtained a criterion for the completeness of the system of functions of k -valued logic with respect to the superposition operation.

The need for representations of functions of n variables in the form of superposition of functions of a smaller number of variables arose with the development of the theory and practice of neural networks. The emergence of neural networks is associated with the article of McCulloch and Pitts [5], which describes the mathematical model of the neuron and neural network. It has been proved that Boolean functions can be represented by neural networks. Later, F. Rosenblatt [6] proposed a model called perceptron and the training algorithm for this model. It has been shown that perceptrons can solve some tasks more efficiently than computers of traditional architecture. Later, a serious mathematical analysis of perceptrons, carried out by M. Minsky and S. Pipert [7], revealed serious limitations in the field of perceptron application. In particular, it has been shown that solving problems with perceptron requires a significant amount of time or a large number of neurons.

In the future, the restrictions were relaxed by the way of replacing the threshold function of activating neurons to the sigmoid function. In particular, in 1989 G. Cybenko [8], K. Funahashi [9] and K. Hornik [10] independently proved the following fact. Let ψ is a fixed sigmoidal function, and f is

the continuous function on compactum $K \subset \mathbb{R}^n$ of n variables. Then f can be approximated in the sense of uniform approximation of a four-layer network (with two hidden layers), with the activation functions of the first and last layer linear, and for intermediate layers ψ .

Hecht–Nielsen [11] showed that the continuous function of many variables can be represented by a three-layer neural network with n components of the input signal, $2n + 1$ components of a hidden layer with sigmoid activation functions, and M components of the output layer with unknown activation functions and represented by two aggregation operations. Thus, in a nonconstructive form, the solution of the problem of representing a function of n variables by a neural network was proved.

In [12], it is shown that in the integral metric with the weight of Chebyshev–Hermite, the function of n variables can be approximated by a three-layer neural network, with the activation functions of the hidden layer can be given in advance, and the output is linear.

In this paper, we propose a modified three-layer neural network with architecture that has only diagonal synaptic connections between neurons, as a result we obtain the transformed Hecht–Nielsen theorem. Such an architecture of a three-layer neural network ($m = 2n + 1$ is the number of neurons in the hidden layer of the neural network; n is the number of input signals) allows an approximation of a function of n variables, with a given accuracy $\varepsilon > 0$, by means of one aggregation operation.

2. The Formulation of the transformed Hecht–Nielsen theorem

In recent years, there is a significant interest in the hypothesis of the quantum-statistical nature of human consciousness [13]. Therefore, synaptic connections that are built between neurons in artificial neural networks must be described by statistical laws. Weight coefficients of synaptic connections λ_{ik} should be described by average value $\bar{\lambda}_{ik} = \langle \lambda_{ik} \rangle$ and dispersion $\tilde{\lambda}_{ik} = \langle \lambda_{ik}^2 \rangle - \langle \lambda_{ik} \rangle^2$.

Let the random variables λ_{ik} (weighted coefficients of synaptic bonds between i -th and k -th neurons) do not depend on those connections that exist between other neurons $\lambda_{\alpha\beta}$. In this case, the statistical properties λ_{ik} are entirely determined by the distribution function $f(\lambda_{ik})$ which is on the connection between the i -th and k -th neurons. Assume that $f(\lambda_{ik})$ is a Gaussian distribution function

$$f(\lambda_{ik}) = \frac{1}{\bar{\lambda}_{ik} \sqrt{2\pi}} \exp\left(-\frac{\lambda_{ik}^2}{2\tilde{\lambda}_{ik}}\right), \quad (1)$$

which is given by two parameters: average value $\bar{\lambda}_{ik} = \langle \lambda_{ik} \rangle$ and dispersion $\tilde{\lambda}_{ik} = \langle \lambda_{ik}^2 \rangle - \langle \lambda_{ik} \rangle^2$.

Kolmogorov's theorem on the representation of a continuous function of n variables in the form of a superposition m of continuous functions of one variable is considered to be a mathematical justification for the use of an artificial neural network for approximating the function of many variables through a linear combination of functions of one variable. For an artificial neural network with diagonal and non-diagonal synaptic connections between neurons, this theorem was formulated by Hecht–Nielsen [11]: For an arbitrary $\varepsilon > 0$ there is a three-layer neural network whose output functions for a hidden layer are $g_j(\mathbf{V})$, and the output functions (activation functions) for the input and output layers are linear and have the function of input-output $F_\varepsilon(V_1, V_2, \dots, V_n) = \sum_{j=1}^m W_j \cdot g_j[\sum_{i=1}^n \lambda_{ij}(V_i)]$, such that $\max_{V \in K} |F(V_1, V_2, \dots, V_n) - F_\varepsilon(V_1, V_2, \dots, V_n)| < \varepsilon$, where K is the compact subset (limited closed subset) \mathbb{R}^n , and $F(V_1, V_2, \dots, V_n)$ is a real continuous function on K . That is, any continuous function $F(\mathbf{V})$ from n variables $\mathbf{V} = (V_1, V_2, \dots, V_n)$ can be approximated with a given accuracy $\varepsilon > 0$ by means of a three-layer neural network with one hidden layer, which is described by the equation $F_\varepsilon(V_1, V_2, \dots, V_n) = \sum_{j=1}^m W_j \cdot g_j[\sum_{i=1}^n \lambda_{ij}(V_i)]$, for which it is enough to adjust the $m = 2n + 1$ transfer functions $\lambda_{ij}(V_i)$ of the neurons of the hidden layer; where $\lambda_{ij}(V_i)$ are sensory functions, and g_j are activation functions of the hidden layer neurons. As can be seen from the formulation of the theorem for approximation of the function $F(\mathbf{V})$ of the n variables $\mathbf{V} = (V_1, V_2, \dots, V_n)$, a three-layer neural network uses two aggregation operations $\sum_{j=1}^m \dots \sum_{i=1}^n \dots$.

We formulate the Hecht–Nielsen theorem for a modified three-layer artificial neural network with diagonal synaptic connections between neurons.

Proposition. For an arbitrary $\varepsilon > 0$ there is a three-layer neural network whose output functions for a hidden layer are $g_j(\mathbf{V})$, and the output functions (activation functions) for the input and output layers are linear and have the function of input-output $\tilde{F}_\varepsilon(V_1, V_2, \dots, V_n) = \sum_{j=1}^m W_j \cdot g_j [\sum_{i=1}^n \tilde{\lambda}_{ij}(V_i)]$, such that $\max_{V \in K} |F(V_1, V_2, \dots, V_n) - \tilde{F}_\varepsilon(V_1, V_2, \dots, V_n)| < \varepsilon$. That is, any continuous function $\tilde{F}(\mathbf{V})$ of n variables $\mathbf{V} = (V_1, V_2, \dots, V_n)$ can be approximated with a given accuracy $\varepsilon > 0$ by means of a three-layer neural network with one hidden layer and one aggregation operation, for which it is enough to diagonalize the matrix of synaptic connections between the vector of input signals and the hidden layer of nonlinear neurons and adjust the $m = 2n + 1$ transfer functions $\tilde{\lambda}_{ij}(V_i)$ of the neurons of the hidden layer.

$$\tilde{F}_\varepsilon(V_1, V_2, \dots, V_n) = \sum_{j=1}^m W_j \cdot g_j [\beta_j V_j + \theta_j], \tag{2}$$

where $g_j [\beta_j V_j + \theta_j]$ is the activation function of the j -th neuron of the hidden layer; W_j are weight coefficients of synaptic connections from the hidden layer to the output layer; θ_j is the additive component of the input signal (threshold of excitation of a neuron).

Proof. Let us give the architecture of a classical artificial three-layer neural network (with one hidden layer) (Fig. 1), by which it is possible with a given accuracy $\varepsilon > 0$ we can approximate the continuous function $\mathbf{V} = (V_1, V_2, \dots, V_n)$ for which it is enough to adjust the $m = 2n + 1$ transfer functions $\tilde{\lambda}_{ij}(V_i)$ of the neurons of the hidden layer

$$F_\varepsilon(V_1, V_2, \dots, V_n) = \sum_{j=1}^m W_j \cdot g_j \left[\sum_{i=1}^n \tilde{\lambda}_{ij}(V_i) \right], \tag{3}$$

where $\tilde{\lambda}_{ij}(V_i)$ are sensory functions that establish a functional relationship between the input signal V_i and the weight coefficients of the synaptic connections $\tilde{\lambda}_{ij}$ with the hidden layer neurons.

$$\tilde{\lambda}_{ij}(V_i) \equiv \tilde{\lambda}_{ij} \cdot V_i + \theta_j, \tag{4}$$

where $\tilde{\lambda}_{ij}$ are weight coefficients of synaptic connections from the input to the hidden layer.

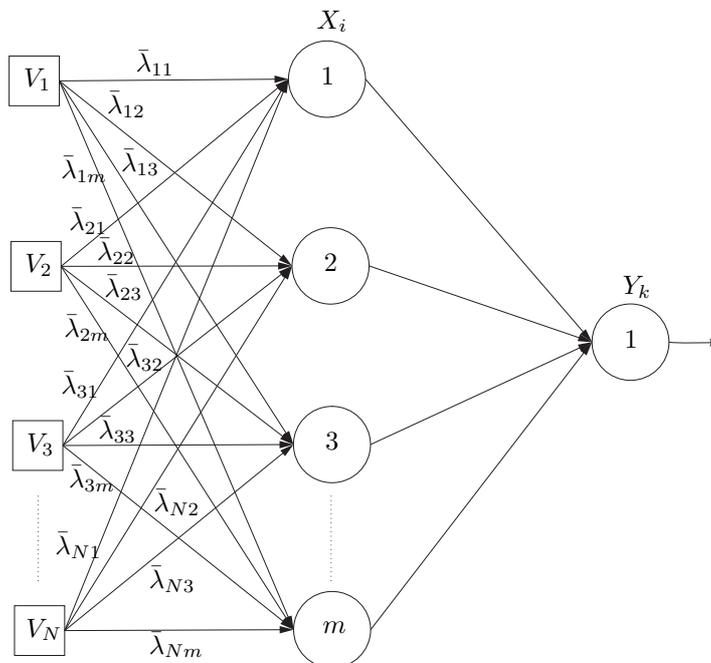


Fig. 1. Schematic representation of a three-layer neural network with diagonal and non-diagonal weighting coefficients of synaptic connections $\tilde{\lambda}_{ij}$.

In order to obtain the architecture of a modified artificial neural network with nonlinear oscillatory neurons [14,15] from the architecture of the classical three-layer neural network (Fig. 1), it is necessary to diagonalize [16] the matrix of synaptic connections $\hat{\lambda}_{ij}$ between the input signals $\mathbf{V} = (V_1, V_2, \dots, V_n)$ and the hidden layer. Such a modified artificial dynamic neural network will be called an artificial neural network with diagonal synaptic connections.

3. Modified neural network with diagonalized synaptic connections

To diagonalize the matrix of synaptic connections and memorize the prototype of the input signal with a three-layer neural network (Fig. 1), we write the input signal as a deterministic vector

$$\mathbf{V} = (V_1, V_2, \dots, V_n), \quad (5)$$

where $V_i = \mathbf{e} \mathbf{V}$ is the projection of \mathbf{V} on \mathbf{e} , (\mathbf{e} is i -th base vector of the coordinate system).

To memorize the prototype of the image (information signal) we impose restrictions on synaptic connections $\bar{\lambda}_{ij}$ (synaptic connections from sources V_1, V_2, \dots, V_n to neurons $1, 2, 3, \dots, N$)

$$\bar{\lambda}_{ij} = V_i \cdot V_j, \quad \bar{\lambda}_{ij} \neq \bar{\lambda}_{ji}, \quad \text{when } i \neq j,$$

and form matrix $\hat{\lambda}_{ij}$ with deterministic matrix elements

$$\langle \bar{\lambda}_{ij} \rangle = \langle V_i \rangle \cdot \langle V_j \rangle \quad (6)$$

and bring it to a diagonal form with real own values

$$\tilde{\lambda}_{ij}(V_1, V_2, \dots, V_n) = \beta_i(V_1, V_2, \dots, V_n) \cdot \delta_{ij}. \quad (7)$$

To bring matrix $\hat{\lambda}_{ij}$ to a diagonal [16] form, we reduce it to a symmetric shape and make a linear transformation

$$\hat{\lambda} = \hat{U}^{-1} \hat{\lambda} \hat{U}, \quad (8)$$

where \hat{U} is the matrix consists of its own base vectors \mathbf{u}_j of the matrix \mathbf{u}_j , that is $\hat{U} = (\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_j, \dots, \mathbf{u}_n)$;

$$\hat{\lambda} \mathbf{u}_j = \beta_j \mathbf{u}_j. \quad (9)$$

In the basis of eigenvectors \mathbf{u}_j the matrix of linear transformation $\hat{\lambda}$ has a diagonal form, with the main diagonal there are real eigenvalues of $\tilde{\lambda}_{ij}(V_1, V_2, \dots, V_n) = \beta_i(V_1, V_2, \dots, V_n) \cdot \delta_{ij}$ matrices $\hat{\lambda}$; where δ_{ij} is the symbol of Kronecker; $\beta_i(V_1, V_2, \dots, V_n)$ are valid eigenvalues of the diagonal matrix of synaptic connections.

Substituting in (3), taking into account the form (4), the diagonal elements of the matrix of synaptic connections (7), we obtain the modified Hecht–Nielsen theorem (2). Which describes the architecture of an artificial dynamic three-layer neural network with diagonal synaptic connections and with one aggregation operation (Fig. 2).

It should be noted that diagonalized the matrix of synaptic connections leads to a decrease the aggregation operations ($A_{j=1}^m = \sum_{j=1}^m$ and/or $\prod_{j=1}^m \dots$), which in turn reduces the computing resource and reduces the time to adjust the weighting coefficients of synaptic connections in the neural network during its learning through changing the network configuration by way of excluding non-diagonal elements of synaptic connections between neurons (Figs. 1, 2), or between a vector of output signals of neurons in a hidden layer with the neurons of the output layer.

To memorize N different image-standards (N various prototypes-images) the coefficients of synaptic connections $\bar{\lambda}_{ij}$ are selected in the form

$$\bar{\lambda}_{ij} = \sum_{p=1}^N V_i^{(p)} \cdot V_j^{(p)}. \quad (10)$$

The algorithm for bringing $\bar{\lambda}_{ij}$ (10) to the diagonal form [16] is similar (6).

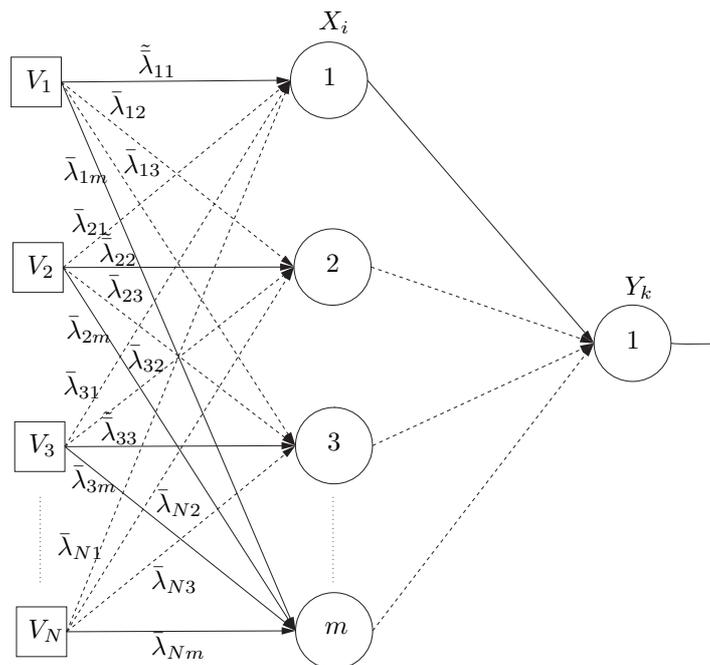


Fig. 2. Schematic representation of a three-layer neural network after diagonalization of the matrix of weight coefficients of synaptic connections $\bar{\lambda}_{ij}$.

The diagonalization operation can be used to simplify the weighting of synaptic connections in the neural network during process of its learning due to the change in network configuration by the way of excluding some existing connections between neurons and reducing the elements of the communication matrix between the vector of input signals and neurons (Fig. 2), or between a vector of output signals of neurons in a hidden layer with the neurons of the output layer.

A comparative analysis of neural networks (Figs. 1, 2) shows that, before the diagonalization of the weight of synaptic bonds, their number was N^2 , whereas after the diagonalization became N [17]. Such a decrease in the number of weight coefficients of synaptic connections results in a seal information which is fed from the input images to the neural network through the channels of the main elements $\bar{\lambda}_{ii}(i = 1, 2, \dots, N)$ of the matrix of synaptic connections between the neurons.

4. Results of numerical experiment

The test problem of the approximation of the function F of two variables V_1, V_2 that is $F(V_1, V_2) = V_1^2 - V_2^2 + V_1V_2$, non-diagonalized and diagonalized three-layer neural networks, described by equations $F_\varepsilon(V_1, V_2) = \sum_{j=1}^5 W_j \cdot \tanh \left[\sum_{i=1}^2 \bar{\lambda}_{ij} \cdot V_i - \theta_j \right]$ and $\tilde{F}_\varepsilon(V_1, V_2) = \tilde{W}_1 \tanh (\tilde{\lambda}_{11} (V_1^2 + V_2^2) - \tilde{\theta}_1)$, is considered. The task of approximating the equivalent of the task of teaching with a teacher.

The problem of approximation with non-diagonalized and diagonalized three-layer neural networks is to adjust the weighted synaptic coefficients λ_{ij} and W_j by the minimization criterion ε :

$$\varepsilon = \frac{1}{2} \left[W_1 \tanh (\bar{\lambda}_{11} V_1 + \bar{\lambda}_{12} V_2 - \theta_1) + W_2 \tanh (\bar{\lambda}_{21} V_1 + \bar{\lambda}_{22} V_2 - \theta_2) + W_3 \tanh (\bar{\lambda}_{31} V_1 + \bar{\lambda}_{32} V_2 - \theta_3) + W_4 \tanh (\bar{\lambda}_{41} V_1 + \bar{\lambda}_{42} V_2 - \theta_4) + W_5 \tanh (\bar{\lambda}_{51} V_1 + \bar{\lambda}_{52} V_2 - \theta_5) - V_1^2 - V_2^2 + V_1 V_2 \right]^2 \rightarrow \min(\bar{\lambda}_{ij}, W_i, \theta_i),$$

where $i = \overline{1, 5}; j = 1, 2$.

$$\varepsilon = \frac{1}{2} \left[\tilde{W}_1 \tanh (\tilde{\lambda}_{ij} (V_1^2 + V_2^2) - \tilde{\theta}_1) - V_1^2 - V_2^2 + V_1 V_2 \right]^2 \rightarrow \min(\tilde{\lambda}_{ij}, \tilde{W}_i, \tilde{\theta}_i),$$

where $i = j = 1$.

Table 1. $F_\varepsilon(V_1, V_2) = \sum_{j=1}^5 W_j \tanh \left[\sum_{i=1}^2 \bar{\lambda}_{ij} \cdot V_i - \theta_j \right]$.

N	V_1	V_2	ε	λ_{11}	λ_{12}	λ_{21}
1	0.01	0.1	10^{-5}	0.10038	0.10387	0.10038
2	0.01	0.3	10^{-5}	0.10446	0.23395	0.10446
3	0.01	0.5	10^{-5}	0.1042	0.31	0.1042
4	0.01	0.7	10^{-5}	0.10387	0.37109	0.10387
5	0.01	0.9	10^{-5}	0.10361	0.4251	0.10361
6	0.1	0.01	10^{-5}	0.10387	0.10038	0.10387
7	0.5	0.01	10^{-5}	0.31	0.1042	0.31
8	0.9	0.01	10^{-5}	0.4251	0.10361	0.4251
9	0.9	0.1	10^{-5}	0.39408	0.13267	0.39408
10	0.5	0.5	10^{-5}	0.19412	0.19412	0.19412
11	0.5	0.9	10^{-5}	0.20529	0.28953	0.20529
12	0.9	0.5	10^{-5}	0.28953	0.20529	0.28953

Table 2.

$$F_\varepsilon(V_1, V_2) = \sum_{j=1}^5 W_j \tanh \left[\sum_{i=1}^2 \bar{\lambda}_{ij} \cdot V_i - \theta_j \right]$$

Table 3.

$$F_\varepsilon(V_1, V_2) = \sum_{j=1}^5 W_j \tanh \left[\sum_{i=1}^2 \bar{\lambda}_{ij} \cdot V_i - \theta_j \right]$$

N	V_1	V_2	λ_{22}	λ_{31}	λ_{32}	λ_{41}
1	0.01	0.1	0.10387	0.10038	0.10387	0.10038
2	0.01	0.3	0.23395	0.10446	0.23395	0.10446
3	0.01	0.5	0.31	0.1042	0.31	0.1042
4	0.01	0.7	0.37109	0.10387	0.37109	0.10387
5	0.01	0.9	0.4251	0.10361	0.4251	0.10361
6	0.1	0.01	0.10038	0.10387	0.10038	0.10387
7	0.5	0.01	0.1042	0.31	0.1042	0.31
8	0.9	0.01	0.10361	0.4251	0.10361	0.4251
9	0.9	0.1	0.13267	0.39408	0.13267	0.39408
10	0.5	0.5	0.19412	0.19412	0.19412	0.19412
11	0.5	0.9	0.28953	0.20529	0.28953	0.20529
12	0.9	0.5	0.20529	0.28953	0.20529	0.28953

N	V_1	V_2	λ_{42}	λ_{51}	λ_{52}
1	0.01	0.1	0.10387	0.10038	0.10387
2	0.01	0.3	0.23395	0.10446	0.23395
3	0.01	0.5	0.31	0.1042	0.31
4	0.01	0.7	0.37109	0.10387	0.37109
5	0.01	0.9	0.4251	0.10361	0.4251
6	0.1	0.01	0.10038	0.10387	0.10038
7	0.5	0.01	0.1042	0.31	0.1042
8	0.9	0.01	0.10361	0.4251	0.10361
9	0.9	0.1	0.13267	0.39408	0.13267
10	0.5	0.5	0.19412	0.19412	0.19412
11	0.5	0.9	0.28953	0.20529	0.28953
12	0.9	0.5	0.20529	0.28953	0.20529

Table 4.

$$F_\varepsilon(V_1, V_2) = \sum_{j=1}^5 W_j \tanh \left[\sum_{i=1}^2 \bar{\lambda}_{ij} \cdot V_i - \theta_j \right]$$

Table 5.

$$\tilde{F}_\varepsilon(V_1, V_2) = \tilde{W}_1 \tanh \left(\tilde{\lambda}_{11}(V_1^2 + V_2^2) - \tilde{\theta}_1 \right)$$

N	W_1	W_2	W_3	W_4	W_5
1	0.10425	0.10425	0.10425	0.10425	0.10425
2	0.23609	0.23609	0.23609	0.23609	0.23609
3	0.31264	0.31264	0.31264	0.31264	0.31264
4	0.37635	0.37635	0.37635	0.37635	0.37635
5	0.43631	0.43631	0.43631	0.43631	0.43631
6	0.10425	0.10425	0.10425	0.10425	0.10425
7	0.31264	0.31264	0.31264	0.31264	0.31264
8	0.43631	0.43631	0.43631	0.43631	0.43631
9	0.41287	0.41287	0.41287	0.41287	0.41287
10	0.25752	0.25752	0.25752	0.25752	0.25752
11	0.34885	0.34885	0.34885	0.34885	0.34885
12	0.34885	0.34885	0.34885	0.34885	0.34885

N	V_1	V_2	ε	$\tilde{\lambda}_{11}$	\tilde{W}_1
1	0.01	0.1	10^{-5}	0.76674	0.76675
2	0.01	0.3	10^{-5}	0.9658	0.96702
3	0.01	0.5	10^{-5}	0.98844	0.99852
4	0.01	0.7	10^{-5}	1.00808	1.04938
5	0.01	0.9	10^{-5}	1.03788	1.16307
6	0.1	0.01	10^{-5}	0.76674	0.76675
7	0.5	0.01	10^{-5}	0.98844	0.99852
8	0.9	0.01	10^{-5}	1.03788	1.16307
9	0.9	0.1	10^{-5}	0.98185	1.09018
10	0.5	0.5	10^{-5}	0.70973	0.72467
11	0.5	0.9	10^{-5}	0.79113	0.88602
12	0.9	0.5	10^{-5}	0.79113	0.88602

The optimization problem is solved by the gradient method with an error of ($\varepsilon = 10^{-5}$). The results of this test task are given in Tables 1–5.

5. Conclusions

A transformed Hecht–Nielsen theorem is obtained based on a modified three-layer artificial neural network with diagonalized synaptic connections between neurons. This theorem describes an algorithm for approximating the function of n variables, with given accuracy $\varepsilon > 0$, by means of one aggregation operation.

It is shown that the diagonalization of the matrix of synaptic connections in a three-layer neural network leads to one aggregation operation, which, in turn, reduces the computing resource and the time of adjustment of the weight coefficients of synaptic connections in the neural network during its learning due to changing the network configuration by eliminating non-diagonal elements synaptic connections between neurons.

The experiment has shown that for the same error of calculation ($\varepsilon = 10^{-5}$) the test problem of approximation of the function F that uses a three-layer diagonalized neural network requires less synaptic connections (2 synaptic connections, namely $\bar{\lambda}_{11}$ and \bar{W}_1) than a non-diagonalized three-layer neural network (15 synaptic connections $\bar{\lambda}_{ij}, W_i$, where $i = \overline{1, 5}, j = \overline{1; 2}$).

-
- [1] Kolmogorov A. N. On the representation of continuous functions of several variables by superpositions of continuous functions of a smaller number of variables. DAN USSR. **108**, 2 (1956).
 - [2] Kolmogorov A. N. On the representation of continuous functions of several variables by superpositions of continuous functions of one variable and addition. DAN USSR. **114**, 953–956 (1957).
 - [3] Vitushkin A. G., Khenkin G. M. Linear superposition of functions. Russ. Math. Surv. **22**, 77–125 (1967).
 - [4] Slupecki J. The criterion of completeness of multivalent accounting system. C. R. Classe III. **32** (1–3), 102–109 (1939).
 - [5] McCulloch W. S., Pitts W. A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics. **5** (4), 115–133 (1943).
 - [6] Rosenblatt F. Principles of Neurodynamics: Perceptrons and the theory of brain mechanisms. Spartan Books. Washington, DC (1961).
 - [7] Minsky M., Papert S. Perceptrons. Oxford, England: M.I.T. Press. (1969).
 - [8] Cybenko G. Approximations by superpositions of sigmoidal functions. Math. Control Signals Systems. **2**, 303–314 (1989).
 - [9] Funahashi K. On the approximate realization of continuous mappings by neural networks. Neural Networks. **2** (3), 183–192 (1989).
 - [10] Hornik K., Stinchcombe M., White H. Multilayer feedforward networks are universal approximators. Neural Networks. **2** (5), 359–366 (1989).
 - [11] Hecht–Nielsen R. Kolmogorov’s mapping neural network existence theorem. IEEE First Annual Int. Conf. on Neural Networks, San Diego. **3**, 11–13 (1987).
 - [12] Alekseev D. V. Approximation of functions of several variables by neural networks. Fundamental and applied mathematics. **15** (3), 9–21 (2009).
 - [13] Penrose R. Shadows of the Mind: A Search for the Missing Science of Consciousness. Oxford University Press, Inc. New York, NY, USA (1994).
 - [14] Lytvyn V., Vysotska V., Peleshchak I., Rishnyak I., Peleshchak R. Time dependence of the output signal morphology for nonlinear oscillator neuron based on Van der Pol model. International Journal of Intelligent Systems and Applications. **4**, 8–17 (2018).
 - [15] Peleshchak R. M., Lytvyn V. V., Peleshchak I. R. Dynamics of a nonlinear oscillatory neuron under the action of an external non-stationary signal. Radioelectronics, computer science, management. **4**, 97–105 (2017).
 - [16] Lytvyn V., Peleshchak I., Peleshchak R. The compression of the input images in neural network that using method diagonalization the matrices of synaptic weight connections. 2017 2nd International Conference on Advanced Information and Communication Technologies (AICT). 66–70 (2017).

- [17] Lytvyn V., Peleshchak I., Peleshchak R. Increase the speed of detection and recognition of computer attacks in combined diagonalized neural networks. 2017 4th International Scientific-Practical Conference “Problems of infocommunications. Science and Technology”. 152–155 (2017).

Теорема Хехт–Нільсена для модифікованої нейронної мережі з діагональними синаптичними зв'язками

Пелещак Р.¹, Литвин В.², Пелещак І.², Дорошенко М.¹, Оливко Р.²

¹Дрогобицький державний педагогічний університет ім. І. Франка,
вул. І. Франка, 24, Дрогобич, 82100, Україна

²Національний університет “Львівська політехніка”,
вул. С. Бандери, 12, Львів, 79013, Україна

У роботі запропоновано модифіковану тришарову нейронну мережу з архітектурою, яка має тільки діагональні синаптичні зв'язки між нейронами, внаслідок чого отримано трансформовану теорему Хехт–Нільсена. Така архітектура тришарової нейронної мережі ($m = 2n + 1$ – кількість нейронів прихованого шару нейронної мережі, n – кількість вхідних образів) дає змогу апроксимувати функцію від n змінних із заданою точністю $\varepsilon > 0$ за допомогою однієї операції агрегування. Тришарова нейронна мережа, яка має як діагональні, так і недиагональні синаптичні зв'язки між нейронами, апроксимує функцію від n змінних за допомогою двох операцій агрегування. Крім цього, діагоналізація матриці синаптичних зв'язків приводить до зменшення обчислювального ресурсу і відповідно до зменшення часу налаштування вагових коефіцієнтів синаптичних зв'язків під час навчання нейронної мережі.

Ключові слова: нейронна мережа, діагоналізація матриці, операція агрегування, апроксимація функції.

2000 MSC: 68T01, 68T05, 68T10, 68T15

УДК: 004.81, 004.891.2, 004.891.3