DOI: https://doi.org/10.26565/2304-6201-2024-64-07

УДК (UDC) 519.7, 004.8

Novikov Artem *PhD student, Department of Artificial Intelligence Systems*
Karazin Kharkiv National University, Svobody Sq 4, Kharkiv, Ukraine, 61022
email: artem.slick@gmail.com;
https://orcid.org/0009-0004-5914-7098

Smyrnov Vadym *Master's student, Department of Artificial Intelligence Systems*
Karazin Kharkiv National University, Svobody Sq 4, Kharkiv, Ukraine, 61022
e-mail: vadym.smyrnov@student.karazin.ua;
https://orcid.org/0009-0000-1743-8541

Yanovsky Volodymyr *Doctor of physical and mathematical sciences; Professor*
"Institute for Single Crystals" of National Academy of Sciences, Nauky ave. 60, Kharkiv, Ukraine, 61001
e-mail: yanovsky@isc.kharkov.ua; **https://orcid.org/0000-0003-0461-749X**

Fractal properties of neural networks

The work is devoted to the research of neural networks' properties, which have been extremely intensively used in various applied directions recently. The study of their general and fundamental properties is becoming more and more **actual** due to their wide application.

The key goal of the work is to investigate the reaction field of an artificial neural network in the space of all possible input signals of a certain length. Based on the example of a simple perceptron, zones where the reaction field of the neural network has a structurally complex nature are studied.

Research methods: To research an output signal field, software was developed, which allowed modeling and visualization of the output signal field over the space of all input signals. The software also allowed changing of activation functions, weights, and thresholds of each neuron, which made it possible to research the influence of all these factors on the structural complexity of the output signal field.

As a result, the study established that, in general, within the space of input signals, there are shadow zones where the response field of the neural network has a self-similar fractal structure. Conditions for the appearance of symmetry of such structures were determined, the influence of activation functions, weights and thresholds of network neurons on the properties of fractal structures was investigated. It was revealed that the input layer of neurons predominantly influences these properties. Dependences of the fractal dimension of the structures on the neuron weights were obtained. Changes occurring with the increase in the dimensionality of the input signal space were discussed.

The presence of shadow zones with a fractal output signal field is important for understanding the functioning of artificial neural networks. Such shadow zones define regions within the input signal space where the neural network's response is extremely sensitive even to minute changes in input signals. This sensitivity leads to a fundamental change in output signals with a slight change in input signals.

Keywords: *artificial neural network, space of input signals, field of output signals, perceptron, artificial neuron, fractal structures, fractal dimension.*

Як цитувати: Novikov A. O., Smyrnov V. M., Yanovsky V. V. Fractal properties of neural networks. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління*. 2024. вип. 64. С.66-79. https://doi.org/10.26565/2304-6201-2024-64-07

How to quote: A. O. Novikov, V. M. Smyrnov and V. V. Yanovsky, "Fractal properties of neural networks", *Bulletin of V.N. Karazin Kharkiv National University, series "Mathematical modelling. Information technology. Automated control systems*, vol. 64, pp. 66-79, 2024. https://doi.org/10.26565/2304-6201-2024-64-07

1 Introduction

The emergence of artificial neural networks begins with the work of Warren McCulloch and Walter Pitts [1], in which they proposed the concept of an artificial neuron and artificial neural networks based on electronic circuits. The artificial neuron itself was a simplified abstraction of the understanding of the natural neuron's functioning that existed at the time. Further development of these ideas led to the creation of a perceptron in the 1960s by Frank Rosenblatt [2, 3], which is a classical neural network that solves the problem of dividing signals into two classes. Initially, the perceptron sparked high hopes for artificial intelligence, bolstered by Rosenblatt's statements at a conference [4] in 1958. However, it quickly became clear that the perceptron struggled to discern certain patterns. A particularly significant decline in interest in the perceptron occurred after the publication of M. Minsky and S. Papert's book [5], which criticized the single-layer perceptron.

For an extended period, neural networks remained outside of active scientific research. Interest in this direction was reignited with the realization that neurons of forward-propagation networks with two or more layers have extraordinary power. The situation changed in the 1980s due to the contributions of John Hopfield [6, 7, 8], who proposed a new perspective on the architecture of neural networks and introduced the ideas of feedback and self-learning algorithms for the first time.

Research on algorithms for linear classification continues even now. For example, the paper [9] proposed a classification algorithm that combines the perceptron algorithm with the Helmbold and Varmuth "leave one output" method.

An overview of existing error limits and new limits for the perceptron algorithm or the perceptron core can be found in [11]. Discussion of the obtained limits go beyond standard margin-loss type bounds, allow for any convex and Lipschitz loss function. The development of information technologies in the 2000s showed a lot of applications of neural networks, such as voice and image recognition [12, 13, 14]. Nowadays, research in the field of neural networks has gained immense popularity [15]. However, the focus on practical applications of neural networks diverted attention from their intrinsic properties, which are not studied enough.

This work delves into the structural properties of neural networks' reactions using a simple two-layer perceptron as an example. The primary focus of this article lies in exploring the properties of the output words' field within the space of all input words of a certain length. It is shown that within the space of input words, under general conditions, there are shadow zones where the field of output words has a fractal structure. The range of scales covered by self-similar fractal patterns is determined by the input words' lengths. The study also uncovers the influence of neuron activation functions and their wights on the fractal dimension of these structures, pinpointing a more significant impact from neurons in the first layer. Typical dependences between changes in the fractal dimension and neuron weights are established.

The obtained results regarding the study on artificial neural network behavior's fractal properties hold theoretical and practical significance for enhancing the stability of neural networks and mitigating the impact of minor input data changes that might otherwise obscure the correct output. Research in this direction can open new prospects for neural networks development and provide deeper understanding and improvement of their functionality.

2 Formulation of the problem

Let's consider the properties of input word processing by a simple two-layer perceptron with two inputs and one output. The input words that this perceptron will process are formulated in some finite alphabet $A = (a_1, \dots, a_n)$. Thus, words in the alphabet A will be the inputs of the perceptron. One of the main characteristics of input words, for example $a_1a_3a_2a_1$, is their length $|a_1a_3a_2a_1| = 4$ or the number of letters in a word. After input sequences processing, we will receive a sequence in some alphabet as the output word of the perceptron. For the sake of simplicity, let's assume that the alphabet of the output signals coincides with the alphabet A. Thus, after processing of two input words with a length of n we will receive an answer or a word of length n in the alphabet A as an output of the neural network. Next, we will explore how the neural network's responses to all possible input words look like.

If we consider words of length n in the alphabet of k letters, then k^n words can be created. Then if we take a unit segment and divide it into k^n segments of length $\frac{1}{k^n}$, one word can be assigned to each of the segments. That is, we can enter specific coordinates – placing all words (sequences) in lexicographic order and assigning them to segments of length $\frac{1}{k^n}$ starting from the first left one (see Fig. 2.1). From now on, we will use such coordinates to identify input words.

Then all input words which are applied to the input of the perceptron correspond to a unit square divided into $k^n \times k^n$ squares with a side length of $\frac{1}{k^n}$. Each pair of input words supplied to the neural network's inputs has corresponding coordinates along the axis x and the axis y. The input word given to the input y, defines the coordinate along the axis y and the word given to the input x, defines the coordinate of the pair along the axis x. Thus, each pair of input words corresponds to one specific square with a side length of $\frac{1}{k^n}$. All output words can be assigned to such coordinates in a completely similar way. This unit square can be considered as an input word space of the perceptron with two input channels. All output words can be visualized by assigning a specific "color" to each small square based on a received response to the corresponding input words. Thus, the output word space of the perceptron will be represented as a painted unit square. If input words have infinite length, then each dot of the square defines a particular input word, and its color represents a corresponding output word. The details of such divisions are given in Appendix A.

Next, we will use $A = (0,1,2\ldots,9)$ decimal symbols as the alphabet. All neurons have certain weights, which will change in different experiments. All activation functions will be step functions with values defined in the alphabet A. It is vital to receive a network response in the same alphabet. Thus, we will further consider perceptrons with different activation functions. We will be interested in the structure of all output words over all input words of a certain length, as well as the rearrangement of output words as the length of input words increases. In another words, we will explore the field of output words in the space of input words.

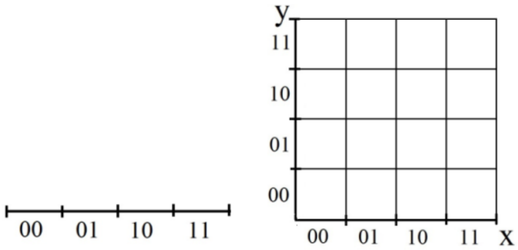

Fig. 2.1. On the left is a division of the unit segment that corresponds to all words of length 2, for simplicity, in the binary alphabet $(0,1)$. *Words are arranged in lexicographic order from left to right. On the right is a unit square with the corresponding coordinates. Each small square corresponds to a certain value of input sequences* (x, y) *of a certain length. Similarly, the unit segment is divided into a larger number of parts that correspond, for example, to words of length 4 and the corresponding coordinates.*

3 Perceptron model

To analyze the properties, a simple model of a two-layer perceptron was built (see Fig. 3.1). It receives input words (x, y) and forms output words z which length matches with the length of the input words. Each neuron of the perceptron weights ω_x, ω_y and a threshold or bias b that can be changed. As discussed before, the values of the activation function correspond to the values of the alphabet $A = (0,1,2\ldots,9)$. The domain of the activation function is all real number. Thus, the general activation function that satisfies the necessary conditions is

$$f(x) = \begin{cases} a_{i_1} & if \quad x < x_1 \\ a_{i_2} & if \quad x_1 \le x < x_2 \\ a_{i_3} & if \quad x_2 \le x < x_3 \\ a_{i_4} & if \quad x_3 \le x < x_4 \\ a_{i_5} & if \quad x_4 \le x < x_5 \\ a_{i_6} & if \quad x_5 \le x < x_6 \\ a_{i_7} & if \quad x_6 \le x < x_7 \\ a_{i_8} & if \quad x_7 \le x < x_8 \\ a_{i_9} & if \quad x_8 \le x < x_9 \\ a_{i_{10}} & if \quad x_9 \le x \end{cases}, \qquad (1)$$

where $a_{i_j} \in A$ are the letters of the alphabet A,
x_i satisfy the conditions $x_1 < x_2 < x_3 < x_4 < x_5 < x_6 < x_7 < x_8 < x_9$ and form ten subintervals.

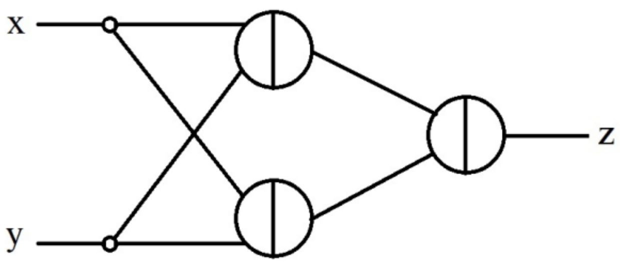

Fig 3.1. The model of the perceptron, which is researched in the work.

Each activation function is defined by a value $x_1, \dots, x_{10}$ and a word $a_{i_1} a_{i_2} \dots a_{i_{10}}$ in the alphabet A. That is, with the given values $x_1, \dots, x_{10}$, we can create 10^{10} different activation functions. It is clear that only a small number of activation functions can be considered in the work. It should be noted that activation functions of different neurons of the perceptron may be different. Additionally, for each neuron, the activation function can divide its domain into intervals of different lengths depending on the choice of weights and threshold of the neuron. Since the argument of the activation function can be values from the interval $[\sigma; (\omega_x + \omega_y)9 + \sigma]$, it is necessary to choose such values of weights, threshold, and type of activation function that the intervals formed by the activation function are close to the interval $[\sigma; (\omega_x + \omega_y)9 + \sigma]$.

Using the developed application, it is possible to choose activation functions, weights, thresholds of neurons, and the length of input words to study the corresponding perceptron response fields. A detailed description of the software can be found in Appendix B.

4 Results of modeling

Fig. 4.1 shows the results of modeling with certain activation functions and neuron weights. On the left is the response field for all input words of length 3, which demonstrates the complex geometric structure of the response field. The part of the answers, which is marked with a blue square, is shown on the right in 10 times enlarged view. Comparing the pictures, it is easy to see the complex, self-similar structure of the answers, which persists as the length of the input words increases. Such structures belong to fractal structures, which have been intensively studied in mathematics and physics (see, for example, [16]). Fractal structures are characterized by a fractal dimension. Having already used this self-similarity, it is possible to calculate the fractal dimension, for example, of black areas [17]. From Fig. 4.1, we can see that the number of black squares is $N = 556688$ with a word length of 3.

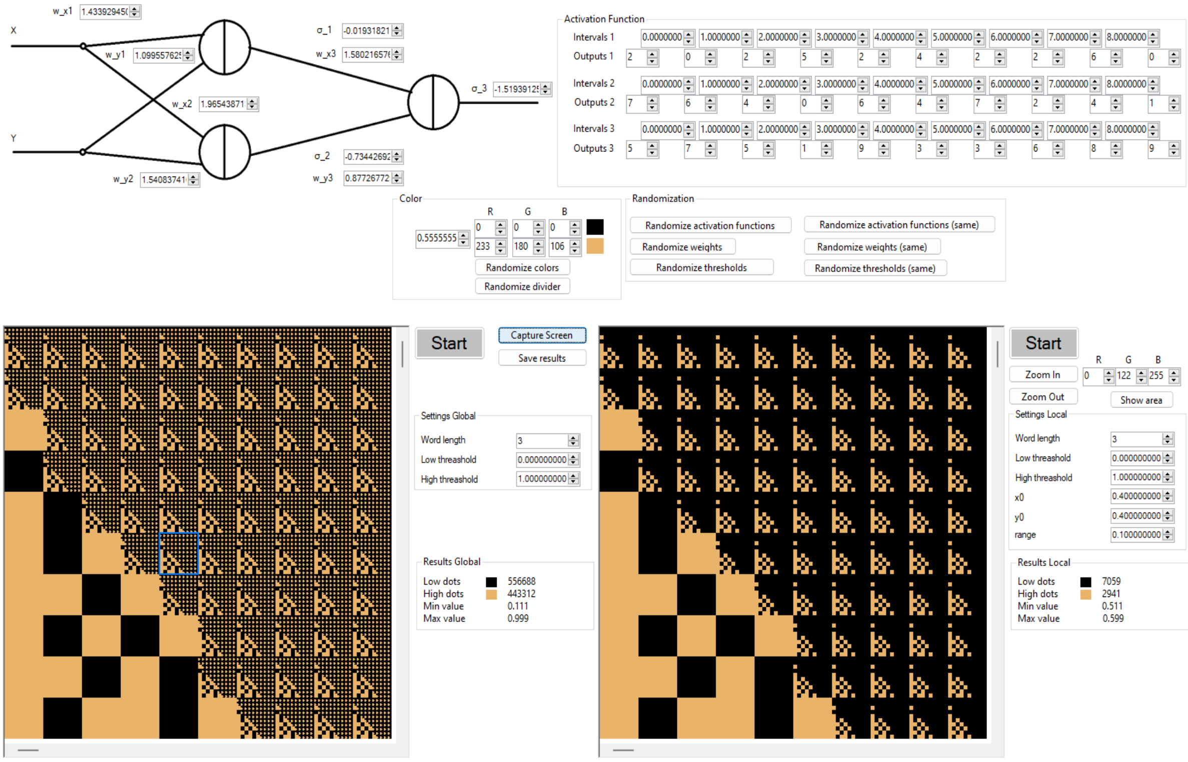

Fig 4.1. An example of a graphic display of output signals with set weights of synapses, for all questions of length 3. On the right, in the upper corner, the activation functions of neurons are shown. Each neuron has its own activation function. The area bounded by the blue square is 10 times enlarged and shown on the left. The comparison of these images shows a self-similar structure on smaller scales. Such self-similarity extends to scales of 10^{-n}, the display of which is impossible for large n due to the limitation of the pixel size.

Thus, by counting the number of black squares N at different input word lengths L, we obtain a linear dependence of $\ln(N)$ on $\ln(\frac{1}{L})$, as shown in Fig. 4.2. Such a linear dependence indicates the fractal structure of the perceptron response field, the fractal dimension of which can be calculated as follows

$$D_F = \frac{\ln(N_2) - \ln(N_1)}{\ln(\frac{1}{L_2}) - \ln(\frac{1}{L_1})},$$

where N_1 and N_2 are the number of black squares with sizes L_1 and L_2, respectively.

That is, in this case, $D_F = \frac{\ln(21563200)-\ln(83)}{\ln(\frac{1}{0,0001})-\ln(\frac{1}{0,1})} \approx 1.805$ is the fractal dimension of the set of a certain answer, marked in black and shown in Fig. 4.1. Such self-similarity is observed for the scales, which are determined by the length of the input words and covers the range of scales from 1 to $\frac{1}{10^n}$, where n is the length of the questions. That is, it occupies a gigantic range of scales. As the length of the words increases, this range increases. Thus, the existence of fractal regions of responses in a simple perceptron is shown.

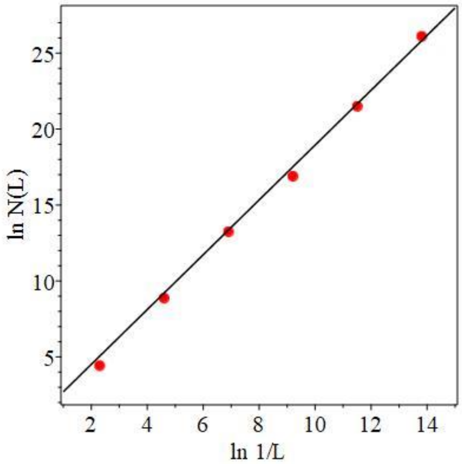

Fig. 4: Graph of the logarithm of the number of black squares N versus the logarithm of the inverse of the side length of one square L. As you can see, a linear relationship is formed, which indicates the presence of a fractal structure the fractal dimension of which is equal to the slope of this line.

It should also be noted that a fractal structure can occur when you select thresholds for the output signals to be displayed that have several decimal places other than 0 that is not less than the length of the output words. For example, if the output words have a length of 3, then a threshold of 0.5555 can be used to create a fractal structure.

Otherwise, the fractality will disappear because when comparing the output signals with the threshold, it is enough to compare only a part of the characters of the output word. For example, if the threshold for displaying output signals is 0.5 and the length of the output words is 3, it is enough to know only the first character of the output word to determine whether the signal exceeds the threshold, which eliminates the need to determine the subsequent response characters. This leads to the elimination of the sensitivity of the structure of the field of output signals to the length of words, which leads to the elimination of fractality.

The presence of such fractal structures in the input word space is important, so changing even the last letter of the input word drastically changes the answer. In other words, there are shadow zones of input words in which the answer fundamentally changes when one last letter in the question is changed. So, for example, if we use the binary alphabet of answers $A = (\text{Yes}, \text{No})$, then when changing the last letter in questions of length 10, instead of the answer "Yes" we can get "No". Such sensitivity will always occur in shadow zones. By shadow zones we will understand the areas of input words, in which the field of output words has a fractal structure. Such a sensitivity can be used for various purposes. Similar shadow zones and fractal patterns of responses were observed for all tested activation functions.

Let us consider the question of what types of structures can be observed with different activation functions and weights of the neurons. It is clear that it is impossible to model perceptrons with all possible activation functions due to their gigantic number. Therefore, it is necessary to consider some of them, based on theoretical considerations and observations for a number of selected activation functions that were used in the simulation.

4.1 Effect of activation functions

The question then arises, what response patterns can be expected for different activation functions? Let's start with the symmetry of the structures. It is easy to understand that if you use neurons that have the property of giving the same response when x is replaced by y, then the field of responses will be symmetrical about the unit square diagonal. An example of a symmetrical structure is shown in Fig. 4.2. In general, such symmetry is absent, as shown in Fig. 4.1. The condition when such symmetry is present limits the properties of activation functions of the first layer neurons only. The activation functions of the f_1 of the first and f_2 of the second neuron of the first layer must satisfy the condition

$$\begin{cases} f_1(\omega_1 x + \omega_2 y + b) = f_1(\omega_1 y + \omega_2 x + b) \\ f_2(\widetilde{\omega}_1 x + \widetilde{\omega}_2 y + b) = f_2(\widetilde{\omega}_1 y + \widetilde{\omega}_2 x + b) \end{cases} \quad (2)$$

which is satisfied by the equality of weights $\omega_1 = \omega_2$ and $\widetilde{\omega}_1 = \widetilde{\omega}_2$ regardless of the form of the activation functions. But values of the weights may be different in the first and second activation functions.

Thus, the symmetry of the structures about the diagonal is fulfilled when the weights of the first layer neurons of both synapses are symmetrical. But there may be other solutions to this functional equation, for special activation functions. An example of symmetrical fractal structures is shown in Fig. 4.3. This figure shows the fulfillment of criteria (2) and, accordingly, the lack of influence of the output layer neuron (that is, its weights and activation functions) on the symmetry of fractal structures.

Now let's discuss the types of fractal structures that arise with various activation functions. During the simulation, it was found that various fractal structures appear for all considered activation functions. This means that the presence of shadow zones and the appearance of fractal structures corresponds to the general case. It is clear that it is impossible to consider all cases of activation functions due to their large number. Various fractal structures were obtained by changing activation functions. An example of one of them is shown in Fig. 4.3. As a result of the simulation, it was found that the appearance and fractal dimension of the structures depend on the choice of the activation function and may change.

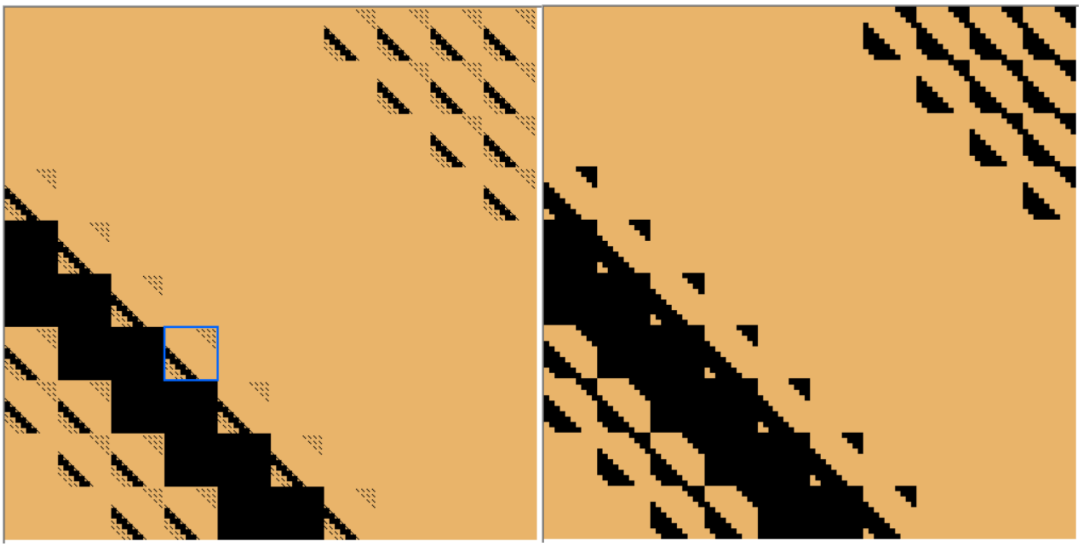

Fig 4.3. An example of a diagonally symmetric self-similar fractal structure. The activation functions of the neurons are different. The first is 6008234346, the second is 9883057840 and the activation function of the output neuron is 6252371258 (see ratio (2)). The first neuron weights are $\omega_x = \omega_y = 0.233$, *the weights of the second neuron are* $\omega_x = \omega_y = 1.348$ *and the third neuron weights are* $\omega_x = 1.692$, $\omega_y = 1.076$. *The threshold of the first neuron is* $\sigma = -0.463$, *the second* $\sigma = -1.339$ *and the third* $\sigma = -1.535$. *Condition (2) is fulfilled. This structure has a fractal dimension* $D_F \approx 1.88$. *Shown on the right is a 10 times magnified portion of the fractal structure drawn in the blue box on the left to demonstrate self-similarity of the output field.*

All described properties are preserved even when choosing the same activation functions for all neurons of the perceptron. The appearance of the fractal structures changes a little, but no qualitative changes are observed.

Thus, the appearance and fractal dimension of the structures change when activation functions are changed.

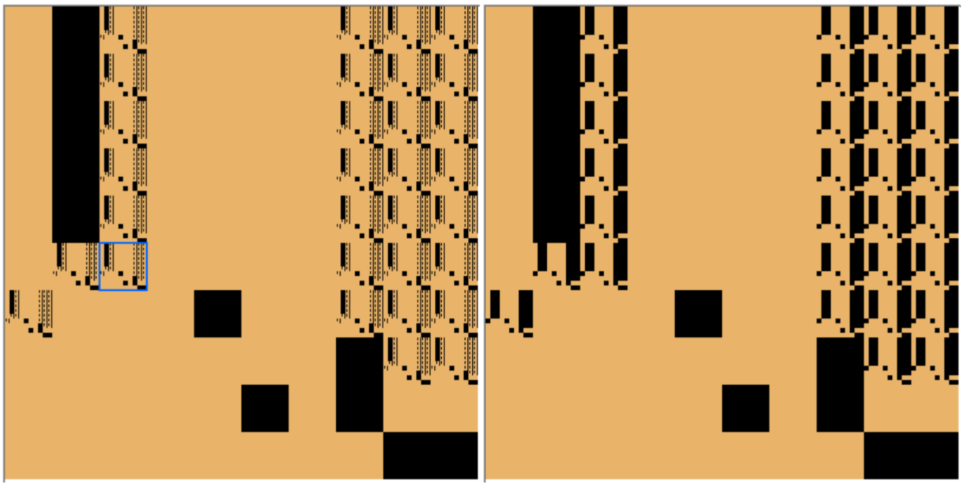

Fig 4.3. Self-similar structure using other activation functions. Activation functions of the neurons are different and defined by the words 2865111974, 2786803702 and 3460777549. Weights of the first neuron are $\omega_x = 1.274$ $\omega_y = 0.088$*, the second neuron weights are* $\omega_x = 0.588$, $\omega_y = 1.775$ *and weights of third neuron are* $\omega_x = 0.823$, $\omega_y = 1.668$*. The threshold of the first neuron is* $\sigma = -0.686$*, the second* $\sigma = -0.195$ *and the third* $\sigma = -0.564$*. Fractal dimension* $D_F \approx 1.85$*. On the right is a 10 times magnified portion of the fractal structure drawn in the blue square on the left to demonstrate self-similarity.*

4.2 Effect of weights of the neurons

Let's consider how alterations of weight values affect fractal structures. It is natural to start with the impact of weights on the symmetry of fractal structures. For instance, changing the weights of neurons, such as those for the symmetric fractal structures mentioned earlier, can break the symmetry about the diagonal. An example demonstrating the loss of symmetry due to alterations of the weight values ω_x of only the first neuron is shown in Fig. 4.5. In the general case of weights and activation functions, such symmetry of fractal structures is absent. Examples of the absence of symmetry under general conditions are shown in Fig. 4.1 and Fig. 4.4.

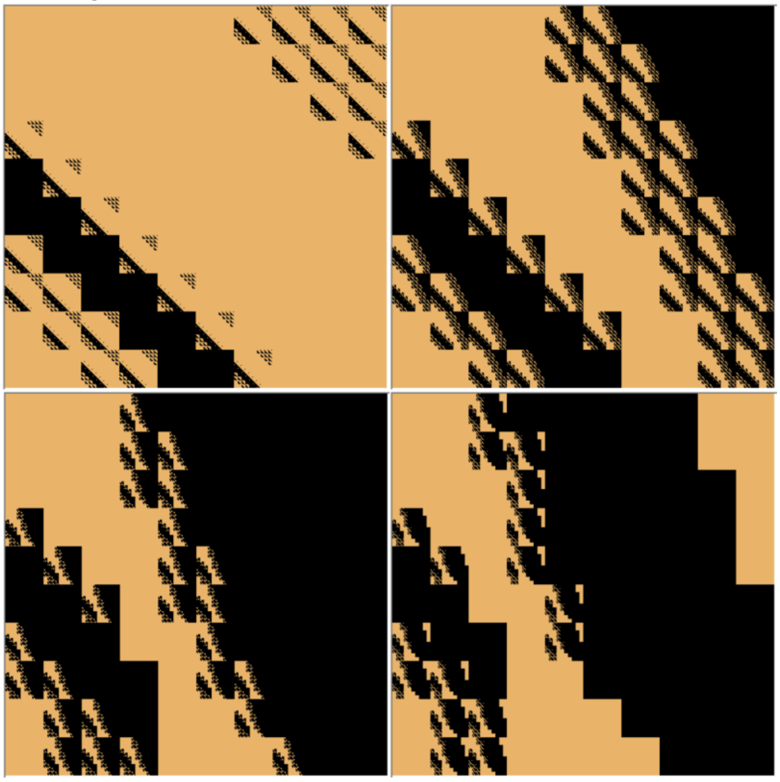

Fig 4.4. In the first figure, a symmetrical structure is generated by a perceptron with activation functions of the neurons defined by the words 6008234346, 9883057840 and 6252371258 and weights of the first neuron $\omega_x = \omega_y = 0.233$*, the second* $\omega_x = \omega_y = 1.348$*, and the third* $\omega_x = 1.692, \omega_y = 1.076$*, and thresholds of the first neuron* $\sigma = -0.463$*, the second* $\sigma = -1.339$*, and the third* $\sigma = -1.535$*. Fractal dimension of the symmetrical*

structure is $D_F \approx 1.871$ *Further, the drawings are obtained by changing only values of the weight* ω_x *of the first neuron. In the following pictures the values of the weight* ω_x *are* 0.253, 0.273 *and 0.293 respectively. Fractal dimensions of the structures* D_F *are* 1.966, 1.984 *and* 1.979 *respectively. They show violation of the symmetry when values of the weight* ω_x *of the first neuron change.*

Moreover, changes in values of the weights of the neurons affect the fractal dimension of the structures as well. Examples of such changes are shown in Fig. 4.4 and Fig. 4.5. The fractal dimensions were found for the structures arising from different values of the weights, showing the dependence between fractal dimensions and the weights of the respective neurons. In each case, only one weight of a certain neuron changed. Figure 4.6 shows the corresponding dependencies. Considering the case of different activation functions of the neurons (see Fig. 4.6, top row), the dependences on the input layer neuron weights are symmetrical. In other words, altering the weights of either neuron results in identical dependences. However, the effects of these neurons show different dependencies. The dependence on the weights of output neuron of the perceptron is more intricate, causing a loss of symmetry in the dependencies (see Fig. 4.6). If every neuron of the perceptron has the same activation function, the dependence of the fractal dimension on the weights is simplified. In the lower row of Fig. 4.6 the left side shows the dependence on the weight of any neuron in the first layer, while the right side shows no changes in the fractal dimension with changes in weights of the output neuron. Consequently, the fractal dimension of the structures within the shadow zones is notably more affected by the weights of neurons in the first layer.

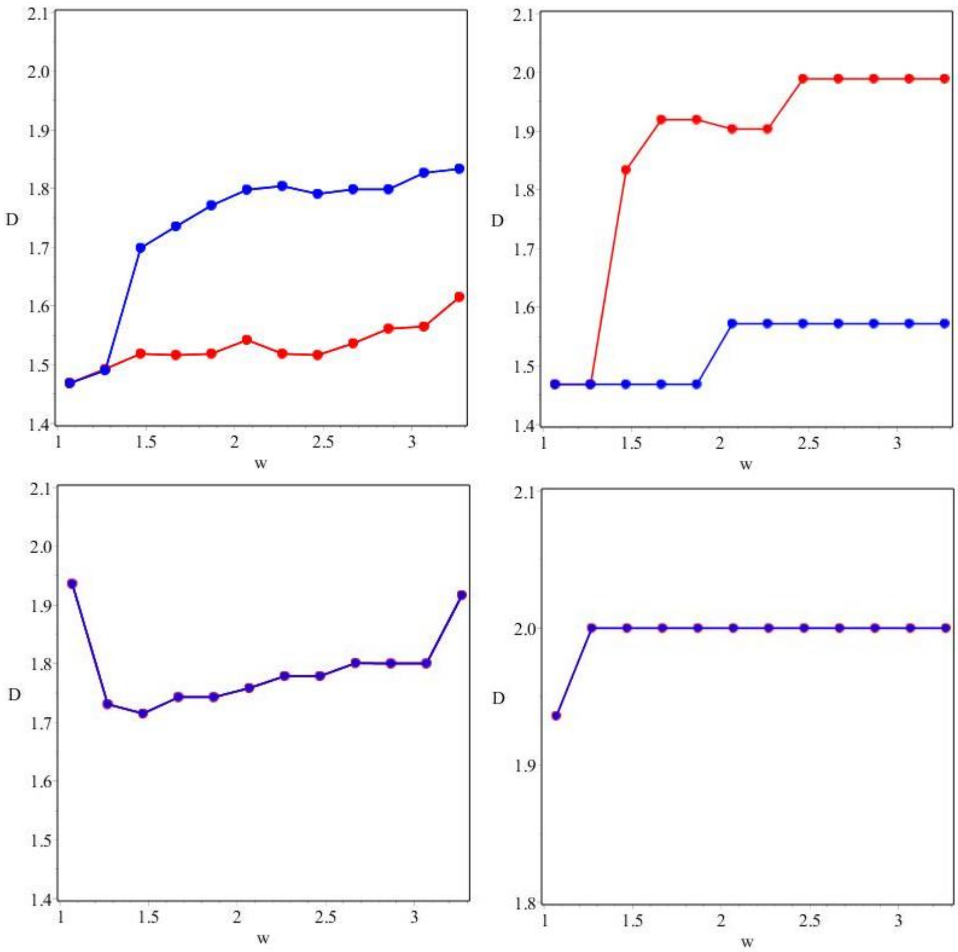

Fig 4.5. The dependence of the fractal dimension on the weights of the neurons of the perceptron. The upper row corresponds to the situation when all activation functions of the neurons are different. The left section stands for the dependence on the weights of the first and second neurons in red and blue, respectively, within the first layer. The right section shows the dependency on the weights of the output neuron, where dependencies on weights ω_x *and* ω_y *are red and blue, respectively. The lower row corresponds to the case of identical activation functions for all neurons. On the left side is the dependency on the weights of any neuron within the first layer, while the right side shows the dependency of the fractal dimension on the weights of the output neuron, which demonstrates its weak influence on the fractal dimension of the structures.*

5 Conclusions

Thus, in the general case, there are shadow zones within the space of input words in which the response field of the perceptron has a fractal structure. Under certain conditions explored in the work, fractal structures can be symmetrical about the diagonal of the unit square. Changing the weights of the neurons of the perceptron within the first layer in a certain way can disrupt this symmetry while the activation function and the weights of the output layer have no impact on the symmetry of fractal structures. In the general case, such structures lack symmetry. The fractal dimension of the structures in the shadow zones within the input word space depends both on activation functions of the neurons and their weights. Changes in the weights of the neurons within the first layer affect this characteristic more significantly, while, in the case of identical activation functions, the weights of the output neuron have a weak influence on its value.

It is clear that similar fractality will occur in more complex neural networks with a larger number of inputs. The main difference is in the large dimension of the space of questions and, accordingly, in the impossibility of demonstrating fractality visually. The dimensionality of the space of input words aligns with the number of input channels, impeding direct visual confirmation of fractality. In this case, verifying fractal characteristics may require specialized data processing algorithms, enabling determination of the fractal dimension without visualization. Such algorithms find application in the study of complex systems such as strange attractors in nonlinear dissipative systems with high dimensionality of the phase space [18]. The alternative approach of their determination involves studying certain intersections between the space of input words and the response field, focusing on these intersections of smaller dimensions.

6 Appendix A

Let's discuss some details of the location of the input words in the unit square and their coloring, according to the received output words. For the sake of simplicity, we will use the binary alphabet $A = (0,1)$. In Chapter 2, we discussed the way to assign coordinates to every input of the perceptron. Now we will discuss the changes that occur as the length of the questions increases. Figure 6.1 shows a comparison of the location of two divisions with different sequence lengths.

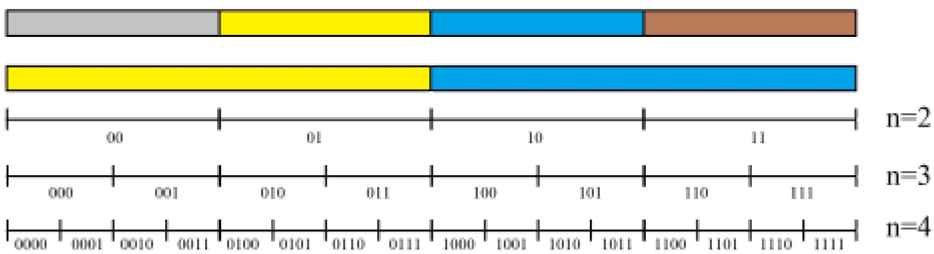

Fig 6.1. Examples of dividing a unit segment for different sentence lengths. The coordinates of the corresponding segments are defined. Below, the length of sentences is 4. It is easy to compare the consistency of a smaller-scale division with a larger-scale one. There are two examples of color palettes shown above that match the corresponding source sentences below, regardless of sentence length.

Such coordinates ensure consistency between the smaller-scale division and the larger-scale one. For instance, if we simplify the corresponding coordinates by truncating with $n = 4$, that is, discarding the last letter, both segments and their coordinates would coincide with the truncation with $n = 3$.

Now let's discuss the visualization of responses to all possible input words. As suggested in Chapter 2, it can be achieved by using some color palette. Any of different palettes can be used for this task. Thus, Fig. 6.1 shows two simple ways of coloring with two colors and 4 colors respectively and the relationship of the palette with words. The color palette stays the same for any division of segments. The color of a word is defined by the color above its position. Of course, it is possible to use more detailed palettes with more colors, including a palette where each word has its own color. Also, in some cases, when a specific sentence must be visualized, a specific color can be assigned to it.

7 Appendix B

To conduct experiments with the perceptron model, we developed a software that allows a user to perform the following actions:

1. setting parameters for the model of the perceptron;
2. calculating output values for different combinations of input signals of the perceptron;
3. setting parameters for graphical representation of results;
4. controlling the calculation process;
5. viewing the result of calculations in the form of a graphic representation of the output signals;
6. viewing the number of squares of a certain color and size to determine the fractal dimension of the constructed image;
7. uploading the results to a file.

The general view of the user interface is shown in Fig. 4.1. Before performing the calculations, all the characteristics of the neural network must be set.

Thus, to set the values of the weights of each connection, the corresponding fields “w_x1”, “w_y1”, “w_x2”, “w_y2”, “w_x3”, “w_y3” must be filled in (Fig. 7.1). To set the threshold values of each neuron, the corresponding fields “σ_1”, “σ_2”, “σ_3” must be filled in (Fig. 7.1).

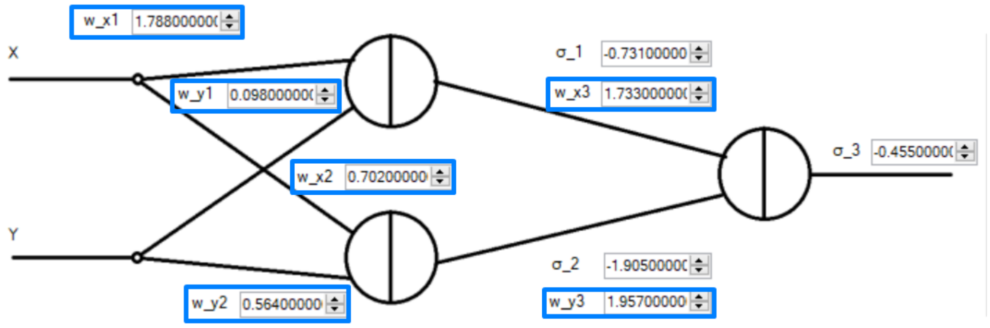

Fig 7.1. Fields for setting the weights of each connection of the perceptron model (highlighted in blue), fields for setting neuron thresholds are not highlighted.

The user can also define activation functions for each neuron. To do this, you need to set the values of the ends of the intervals, in which the corresponding output characters will be set, and the values of these output characters for each interval must be set as well. All these fields are filled in the upper right corner of the interface (see Fig. 4.1).

The software allows the user to construct two images to compare the structure of the global signal field and its parts. To do this, two separate modules were developed. Each of them has a picture box for visual representation of the output field, a “Start” button for starting calculations, “Word length” field to set the length of words, “Low threshold” and “High threshold” fields for setting the minimum and maximum intensity thresholds of the output signals for display, fields “Low dots” and “High dots” for displaying the number of squares painted with the corresponding color, fields “Min value” and “Max value” for displaying the minimum and maximum values of the output signals (Fig. 4.1).

In addition, in the right module for displaying the results there are buttons "Zoom In" and "Zoom Out" to zoom in and out the image, respectively, fields “R”, “G”, “B” to set the color to highlight the part of the global field that is considered on the right, the “Show area” button to select the part of the global field of output signals on the left that is shown on the right, and the “x0”, “y0” and “range” fields to set the initial coordinates and the length of the side of the square within which the output signals of the perceptron are calculated and displayed on the screen (Fig. 4.1).

To set the threshold and color for displaying signals that exceed and do not exceed the set threshold, fill in the appropriate fields in the “Color” section (Fig. 7.2).

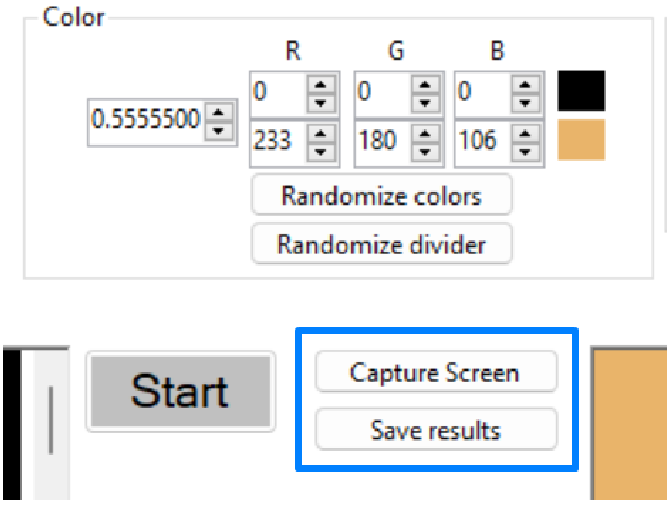

Fig 7.2. Fields for setting colors for displaying low and high signals and a button for saving parameter values and experiment results (highlighted in blue).

After setting all parameters and pressing the "Start" button, the program calculates the values of output signals for all possible combinations of input signals with the specified length in a specified area within the input signal space of the perceptron. The values of the output signals are calculated according to the algorithm described in Chapter 3.

Intensities of the input signals are represented as numbers $0, a_1a_2a_3a_4$, where a_1, a_2, a_3, a_4 are some digits from 0 to 9. In this way, it is possible to analyze all possible values that are transmitted to the inputs of the neural network. Since the neural network must work within a specific alphabet, for simplicity of data processing we will choose the alphabet $A = 0,1,2,3,4,5,6,7,8,9$.

The neural network sequentially processes each symbol of the input. This means that the neuron, which receives the signals $X = 0, x_1x_2x_3x_4$ and $Y = 0, y_1y_2y_3y_4$, processes the symbols x_1 and y_1 on the first iteration, then the symbols x_2 and y_2 are processed etc. Each connection of the perceptron has its own weight. Let the signal X pass through the connection with the weight w_1, and the signal Y – through the connection with the weight w_2, then the total intensity of the signals will be in the interval $D = [b, 9(w_1 + w_2) + b]$.

Since it is necessary to process the signal in such a way as to obtain a symbol from the alphabet A at the output, it is necessary to select the proper activation function. This activation function is the theta-function described in Chapter 3.

After processing all the symbols of the input signals in this way, a sequence of symbols of the output signal will be received. That is, after processing the corresponding symbols of the signals $X = 0, x_1x_2x_3x_4$ and $Y = 0, y_1y_2y_3y_4$, we will get the output signal $Z = 0, z_1z_2z_3z_4$. So, using this algorithm, for each of the possible values of the input signals, the intensities of the output signals are calculated and a matrix of the intensities of the resulting signals is created, where the columns correspond to the values of the first input signal X, and the rows correspond to the values of the second input signal Y. In the intersection of the corresponding column X_n and row Y_k, the value of the output signal $Z_{k,n}$ for the inputs X_n and Y_k is set:

$$\begin{pmatrix} Z_{11} & Z_{12} & \cdots & Z_{1n} \\ Z_{21} & Z_{22} & \cdots & Z_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{k1} & Z_{k2} & \cdots & Z_{kn} \end{pmatrix} \tag{3}$$

Matrix 3 is square, that is $k = n$:

$$\begin{pmatrix} Z_{11} & Z_{12} & \cdots & Z_{1n} \\ Z_{21} & Z_{22} & \cdots & Z_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{n1} & Z_{n2} & \cdots & Z_{nn} \end{pmatrix} \tag{4}$$

After calculations are completed, the matrix (4) of the output values of the perceptron is shown in the form of a square image, in which each point stands for the intensity of the output signal for the corresponding inputs. The specified accuracy of calculations (number of decimal places) is used to construct the image. Each square of the given size is painted with one of the specified colors (Fig. 7.2), depending on whether the signal is greater or less than the set threshold. For example, if you set the word length to 3 and the threshold to 0.555, the output signals from 0.0 to 0.554 will have a color for low signals, and from 0.555 to 0.999 will have a color for high signals. The greater the specified accuracy of the calculations, the smaller the cell size, and therefore the greater their number in the figure.

After calculating the output signals, the software also calculates the number of squares of the corresponding color displayed on the image of the output field of the perceptron to determine the fractal dimension of this field.

In addition, in the results section, the program displays the values of the maximum and minimum signal in the field of output signals represented in the image, as well as the number of colored squares with high signals to verify the calculation of the fractal dimension.

REFERENCES

1. W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bulletin of Mathematical Biophysics*, vol. 5, no. 4, pp. 115–133, 1943. https://doi.org/10.1007/BF02478259
2. F. Rosenblatt, "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain," *Psychological Review*, vol. 65, no. 6, pp. 386–408, 1958. https://doi.org/10.1037/h0042519
3. F. Rosenblatt, *Principles of Neurodynamics*. Washington, DC: Spartan Books, 1962. https://doi.org/10.2307/1419730
4. Olazaran, Mikel, “A Sociological Study of the Official History of the Perceptrons Controversy,” *Social Studies of Science*, vol. 26, no. 3, p.611-659, 1996. https://doi.org/10.1177/030631296026003005
5. M. L. Minsky and S. A. Papert, Perceptrons. Cambridge, MA: MIT Press, 1969. https://doi.org/10.1016/s0361-9230(99)00182-3
6. J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of National Academy of Sciences*, vol. 79, no. 8, pp. 2554–2558, April 1982. https://doi.org/10.1073%2Fpnas.79.8.2554
7. J. J. Hopfield, "Neural with graded response have collective computational properties like those of two-state neurons," *Proceedings of the National Academy of Sciences*, vol. 81, no. 10, pp. 3088-3092, May 1984. https://doi.org/10.1073/pnas.81.10.3088
8. J. J. Hopfield, "Learning algorithms and probability distributions in feed-forward and feed-back networks," *Proceedings of the National Academy of Sciences*, vol. 84, no. 23, pp. 8429-8433, December 1, 1987. https://doi.org/10.1073/pnas.84.23.8429
9. Freund, Y.; Schapire, R. E., "Large margin classification using the perceptron algorithm," *Machine Learning*, vol. 37, no. 3, pp. 277-296, 1999. https://doi.org/10.1023/A:1007662407062
10. Mehryar Mohri1, Afshin Rostamizadeh, “Stability Bound for Stationary Phi-mixing and Beta-mixing Processes,” *Journal of Machine Learning Research (JMLR),* vol. 11, pp. 798-814, 2010. https://doi.org/10.48550/arXiv.0811.1629
11. Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A. R., Jaitly, N. et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine*, IEEE, vol. 29, no. 6, pp. 82-97, 2012. http://dx.doi.org/10.1109/MSP.2012.2205597
12. Jurafsky D., Martin J.H., *Speech and language processing, 2nd edition*. NJ: Prentice Hall, 2008.
13. Shinde, B. S., Dani, A. R., "The origins of digital image processing and application areas in digital image processing medical images," *IOSR Journal of Engineering*, vol. 1, no. 1, pp. 066-071, 2012. http://dx.doi.org/10.9790/3021-0116671
14. He, K., Zhang, X., Ren, S., Sun, J., "Deep residual learning for image recognition," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778, 2016. http://dx.doi.org/10.1109/CVPR.2016.90

15. Schmidhuber, J., "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, no. 3, pp. 85-117, 2015. http://dx.doi.org/10.1016/j.neunet.2014.09.003
16. Benoit B. Mandelbrot, *The Fractal Geometry of Nature*. San Francisco: W.H.Freeman and Co, 1982.
17. Richard M. Crownover, *Introduction to Fractals and Chaos*. University of Missouri-Columbia: Jones and Bartlett Publishers, 1995.
18. Takens, F., "Detecting strange attractors in turbulence," *Dynamical Systems and Turbulence*, vol. 898, pp. 366-381, 1981. https://doi.org/10.1007/BFb0091924

Фрактальні властивості нейронних мереж

Новіков Артем Олександрович — *аспірант кафедри штучного інтелекту та програмного забезпечення Харківський національний університет імені В.Н. Каразіна; майдан Свободи, 6, м. Харків, Україна, 61022*

Смирнов Вадим Максимович — *студент магістратури кафедри штучного інтелекту та програмного забезпечення Харківський національний університет імені В.Н. Каразіна; майдан Свободи, 6, м. Харків, Україна, 61022*

Яновський Володимир Володимирович — *доктор фізико-математичних наук; профессор, Інститут монокристалів, Національна Академія Наук Україн; пр. Науки 60, м. Харків, Україна, 61001*

Роботу присвячено дослідженню властивостей нейронних мереж, які надзвичайно інтенсивно використовуються останнім часом у різноманітних прикладних напрямках. Вивчення їх загальних та фундаментальних властивостей набуває все більшої **актуальності** у зв'язку з широким застосуванням.
Метою роботи є вивчення поля реакції штучної нейронної мережі у просторі всіх можливих вхідних сигналів певної довжини. На прикладі простого персептрона досліджуються зони в яких поле реакцій нейронної мережі має структурно складний тип.
Методи дослідженн: Для дослідження поля вихідних сигналів було розроблено програмне забезпечення, яке дозволило моделювати та візуалізовати поле вихідних сигналів над простором всіх вхідних сигналів. Також програмне забезпечення дозволяло змінювати функцію активації, ваги та пороги кожного нейрону, що дозволило вивчити вплив усіх цих факторів на структурну складність поля вихідних сигналів.
В результаті було доведено, що, у випадку загального положення, у просторі вхідних сигналів існують зони тіні в яких поле реакції нейронної мережі має самоподібну фрактальну структуру. Визначено умови появи симетрії таких структур, досліджено вплив функцій активації, ваг та порогів нейронів мережі на властивості фрактальних структур. Виявлено, що вхідний шар нейронів на ці властивості впливає домінуючим чином. Отримані залежності фрактарної розмірності структур від ваг нейронів. Обговорено зміни, які відбуваються при зростанні розмірності простору вхідних сигналів.
Наявність зон тіні з фрактальним полем вихідних сигналів має важливе значення для розуміння функціонування штучних нейронних мереж. Такі зони тіні визначають області вхідного простору сигналів у яких реакцію нейронної мережі надзвичайно чутлива навіть до незначних змін вхідних сигналів. Це приводить до принципової зміни вихідних сигналів при незначній зміні вхідних сигналів.

Ключові слова: *штучна нейронна мережа, простір вхідних сигналів, поле вихідних сигналів, персептрон, штучний нейрон, фрактальні структури, фрактальна розмірність.*

Надійшла у першій редакції 12.08.2024, в останній - 19.09.2024.
The first version has been received on 12.08.2024, the final version - on 19.09.2024.