



# STOCHASTIC SEARCH SYSTEM FOR PREDICTING DIABETES BASED ON THE GENETIC METHOD

I. Fedorchenko<sup>1</sup>, A. Oliinyk<sup>1</sup>, J.A.J. Alsayaydeh<sup>2</sup>, A Stepanenko<sup>1</sup>, T. Fedoronchak<sup>1</sup>, T. Kolpakova<sup>1</sup> and A. Kharchenko<sup>1</sup>

<sup>1</sup>Department of Software Tools, National University «Zaporizhzhia Polytechnic», Zhukovskoho str., Zaporizhzhya, Ukraine

<sup>2</sup>Fakulti Teknologi Kejuruteraan Elektrik dan Elektronik, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya,

Durian Tunggal, Melaka, Malaysia

E-Mail: [jamil@utem.edu.my](mailto:jamil@utem.edu.my)

## ABSTRACT

The paper considers the field of software engineering related to the parametric synthesis of neural networks based on an evolutionary approach and its application in the diagnosis of diabetes. As a result of the research, an intelligent decision support system for diagnosing diabetes was developed, which is based on machine learning models. The model of parametric optimization of the neural network due to the use of genetic algorithm and particle swarm method is implemented. A modified genetic method for optimizing neural network parameters to solve the problem of predicting the risk of diabetes has been developed. Modification of a simple genetic algorithm, which is implemented as part of the project allows you to speed up the selection of learning parameters of neural networks, and increase the resulting accuracy compared to the basic version of a simple genetic algorithm, by modifying the mutation operator, as well as a changed approach to the selection of individuals for crossing. The developed model is intended for use in the field of medical care and allows to determine with some accuracy the presence of the risk of diabetes in patients by clinical indicators of health. The result of this model is to reduce the likelihood of a doctor's error, increase the doctor's confidence in the decision made in the diagnosis and more lives saved, by making a correct and timely diagnosis. The developed model is intended for use in the field of medical care and allows to determine with some accuracy the presence of the risk of diabetes in patients by clinical indicators of health. The result of this model is to reduce the likelihood of a doctor's error, increase the doctor's confidence in the decision when making a diagnosis and more lives saved, by making a correct and timely diagnosis. The developed model is intended for use in the field of medical care and allows to determine with some accuracy the presence of the risk of diabetes in patients by clinical indicators of health. The result of this model is to reduce the likelihood of a doctor's error, increase the doctor's confidence in the decision when making a diagnosis and more lives saved, by making a correct and timely diagnosis.

**Keywords:** genetic algorithm, classification, neural network, prediction, diabetes, keras, python, PSO.

## INTRODUCTION

To date, there are a huge number of diseases that pose a serious threat to public health, including diabetes. Addressing the diagnosis of diabetes is important because the disease is chronic and can have serious consequences for the entire human body.

Diabetes is a global medical and social problem of today. The prevalence of diabetes is increasing worldwide. According to the World Health Organization, the prevalence is growing rapidly, approaching a global epidemic [1].

Diabetes mellitus has the following consequences: early disability, high mortality rates, measures aimed at rehabilitating patients, lead to high costs for health services. This is due to the fact that the disease and its treatment are determined not from the moment of carbohydrate metabolism, which is determined only by various stress tests (glucose tolerance test and others), but by the manifestation of clinical signs of the disease, which in turn is the cause of various vascular and other tissue changes.

Therefore, it is important to predict the incidence of diabetes using data on human health to plan preventive measures and the corresponding financial costs [1].

## ANALYSIS OF LITERATURE DATA AND PROBLEM STATEMENT

Because diabetes is a chronic disease that is difficult to treat, its diagnosis is an important issue in medical diagnosis. Many scientists around the world are researching this problem, and there are already many predictive models for diagnosing and predicting diabetes, which are both specific (suitable only for predicting / diagnosing diabetes) and general (suitable for many diseases).

In [2], a model developed on the basis of neural networks and a memetic algorithm for predicting diabetes is presented. This model is based on demographic indicators of lifestyle and clinical characteristics. To improve the accuracy of prediction, the authors used a memetic algorithm to update the weights of the neural network. By improving the weights of neural network models, the authors obtained a model of the memetic algorithm, which achieved a model accuracy of 93%. This model is able to provide greater forecasting accuracy than other models discussed in the article.

This study has some limitations. Memetic algorithms require relatively more training time and a more powerful processor than logistic regression models. Also, the amount of data somewhat limited the quality of the study. The model presented in the paper can be



improved. The accuracy of the memetic algorithm can be improved with new data and training. The use of these models by other researchers can improve the reliability of the models and thus save resources. In this study, memetic algorithms were used to update weights and improve the prediction accuracy of the diabetes neural network.

In [3], a system of a fully connected classifier of a multilayer perceptron (BSP) is proposed, which minimizes empirical risk and maximizes erroneously classified points in educational data. In fact, convolutional neural networks develop multiple feature detectors and use them to develop multiple maps, called convolutional layers. During training, the network determines which features are important for the ability to scan input data and for a more accurate classification. Based on this, the classifier itself develops its own characteristics of detectors. In many cases, the functions considered by the neural network will be invisible to the human eye, which is why convolutional neural networks are so useful.

After implementing the CNN-SVM algorithm for classification, it can be concluded that the proposed system can be used for training and model development. This analysis will include the development of a modified neural network with deep learning by integrating the reference vector method. In order to test the system, performance indicators will be compared with an existing method. The proposed system with a modified algorithm significantly optimizes the overall performance of the determination system and effectively reduces the computational complexity of the classifier.

The authors [4] propose a modification of the deep learning coiled neural network due to the integration of the support vector machine. Traditional CNN uses MLP algorithms to classify derived features from raw input. Although these algorithms have shown promising results, MLP algorithms have various shortcomings in the classification of nonlinear datasets. To overcome these shortcomings, the proposed architecture combines the concept of CNN method of selection of features with the SVM classifier. Sets of features from raw data are calculated by mathematically convolving these data with the weights of the Gaussian nucleus. The size of these feature maps using the maximized feature set mapping operation. These optimized feature maps are further classified using SVM with a kernel function for nonlinear data classification.

The modified algorithm significantly optimizes the overall performance of the determination system and effectively reduces the computational complexity of the classifier.

Non-invasive method of diabetes detection is described in [5]. The method includes the implementation of a one-dimensional wrapped neural network for the classification of raw data. The CNN algorithm confuses raw data with a kernel-based filter to create a set of feature maps. The dimensionality of these maps is reduced by using a data sample minimization operation and thereby reducing the sample size. These maps are then applied to a fully connected classifier, which follows a mechanism similar to a multilayer neural network based on a

perceptron. This algorithm reduces the cost of calculations and the need for optimal features.

According to the results, the algorithm performed a successful classification. It is established that the algorithm significantly reduces the root mean square error and optimizes the overall performance of the classifier. In [6], an intelligent prognostic model was proposed that uses in-depth training to predict the risk factor for the disease in patients and the severity of diabetics using a set of conditional data. The model offers in-depth learning in the form of a deep neural network that helps apply predictive analytics to a diabetes data set for optimal results. Existing prognostic models are used to predict the severity and risk factors of diabetics based on data. In the proposed model, the selection algorithm is run for the selection process, then the deep learning module has a deep neural network that uses a limited Boltzmann machine (RBM) as a base unit for data analysis, assigning weight to each branch of the neural network. The test results showed that the number of studies conducted on diabetes using the deep learning model is small. This will help predict diabetes more accurately.

The model can be improved by increasing the data size. In addition, forecasting may depend on other factors that are not present in the specified dataset. Taking these factors into account will further improve the accuracy of forecasting.

The forecasting accuracy with this model can reach 86% depending on the data set.

The authors of [7] built predictive models using the methods of logical regression and gradient amplification (GBM). The plane under the receiver performance curve (AROC) was used to estimate the discriminant ability of these models. The authors used the adjusted threshold method and the weight category method to improve sensitivity - the number of correctly classified patients with diabetes developed by the model. These models were comparable to other machine learning methods, such as the decision tree and the random forest. The results of the models showed themselves as follows: for the gradient gain the sensitivity was 71.6%, for the logical regression model - 73.4%.

In [8], the authors propose a prognostic model called the Improved Diabetes Prediction Algorithm (IDPA), which combines a hierarchical clustering algorithm and a naive Bayesian classification algorithm designed to detect and predict type 2 diabetes. In this case, the IDPA module performs grouping of data into two groups, ie diabetes and non-diabetes, by applying a hierarchical clustering algorithm. Next, filtering is performed by comparing the group value with the class value, followed by the use of a naive Bayesian classifier to predict diabetes. The results show that the proposed method can predict diabetes with a higher level of accuracy (96%) than traditional and existing methods. This model can be used for early prediction of diabetes,

In [9], a hybrid model for predicting the management of type 1 and type 2 diabetes is proposed. The work of the model consists of two stages: autonomous and operational stages. The offline phase focuses on the



preparation of data (structuring) from external sources of knowledge, which are presented in the form of unstructured clinical diagrams and online guidelines for diabetes and the acquisition of knowledge from these sources. Gaining knowledge consists of manual and automatic procedures. In the process guide, first, patient clinical charts are transformed into a structured form called the Diabetes Information System (DMIS) and Guidance on Deviations and Trend Analysis Rules (ATAR). In the offline acquisition phase, a set of rough methods based on the set is used to develop DMPR rules with DMIS. The rules are stored in knowledge bases in the offline process. An online phase is a phase of real-time or model execution that predicts and manages physicians to support their decision-making. This phase is activated when the patient arrives for diagnosis or when the patient registers for follow-up. If the patient is registered for the first time, the HRBR methodology is launched. The HRBR diagnoses with a rough set of considerations (RSR) and predicts the type of diabetes. In case the patient is already registered, only part of the CTA is activated. CTA allows physicians to see all previous patient records in summary form and detect abnormal relationships. The CTA also provides an analysis of future trends for all observations and supports them to identify potential risks in the future. Accordingly, the doctor may take preventive measures.

The method proposed in [10] proposes the combination of KNN and genetic algorithm to improve the accuracy of classification of diabetes data (Figure-2).

The authors [10] used variations of the genetic algorithm to select traits and rank the attributes that are more conducive to classification. The best ranked attributes used to predict the disease and the classification algorithm are based on the evaluated attributes. The classifier is trained to classify data on diabetes, which are both diabetic and non-diabetic. The proposed system is responsible for: selecting the best subset of traits using a genetic algorithm, constructing a classifier and classifier measurement accuracy, and identifying complications associated with diabetes, such as heart and kidney problems, possible stroke, and others.

An optimized CNN (10-layer) model based on DCNN and LeNet-5 was proposed in [11]. This model provides forecasting accuracy of 89-95% (depending on the data set). Changes were made between the third layer of max-pooling and the fully connected layer. In addition to these layers, CNN also includes three convolution layers, three max-pooling layers, and two conjunctival layers. Insufficient number of layers leads to inadequate ability to extract CNN, while, their excessive number increases the time spent on the calculation. In this model, convolution layers are used to extract complex parameters of input feature maps. The max-pooling layers retain the basic features of the input signals while reducing parameters and calculations, which helped to avoid oversize and improve the generalization of the CNN model.

The authors of [12] suggest the use of the sigmoid nucleus and CNN. The originally created SKCNN model

calculates the sigmoid nucleus for disease and drug prediction, drug structural similarity, and disease semantic similarity. The sigmoid core of the drug is combined with the structural similarity of the drugs, and the sigmoid core of the disease is combined with the semantic similarity of the disease. Next, CNN uses a combined similarity of drugs and diseases to obtain functions. In the last step, the Random Forest classifier is introduced to determine whether drugs and diseases are related. The accuracy provided by this model is approximately 92.69%.

The paper [13] presents a new structure of CNN, which is a combination of ideas of two structures Koichi Ito and Gil Levy, where Koichi used deep multitasking, and Levy used the division of the image into five areas. The CNN modification described in the work consists of two parts: the first is four wrap layers with layers of union; the second part is formed of three fully connected layers. Further STL and DMTL are used in the learning process, this method was defined by Koichi Ito. The classification accuracy of this modification is 80.11% when using CNN and the STL training method and 91.34% for CNN and the DMTL training method.

The mathematical dependence of the onset of diabetes on the patient's condition and medical history can be defined as a function in which the possibility of developing the disease will depend on the patient's performance. The generalized dependence model has the form (1):

$$K_{morb} = f(x_n) \quad (1)$$

where  $K_{morb}$  - an indicator that characterizes the risk of diabetes;  $x$  - a set of indicators that characterize the patient's health;  $n$  is the number of indicators.

Based on the above data and analysis of statistical data, we can conclude that the desired mathematical model will not be deterministic, but rather stochastic.

According to the Unified Clinical Protocol of primary and secondary (specialized) medical care [14], the assessment of the risk of developing diabetes in a patient is based on diagnostic measures and a history of the person. The main factors in the development of diabetes include the following indicators:

- the patient's age;
- plasma glucose concentration 2 hours after its introduction in the oral glucose tolerance test;
- serum insulin concentration;
- the value of the diastolic arterial test;
- the patient's body mass index;
- genetics.

Pregnancy and the thickness of the skin fold are also taken into account as additional indicators.

In addition to the above factors influencing the development of diabetes, there are others, but they do not have a significant impact on the occurrence of the disease, so it makes no sense to take them into account.



According to the above indicators of human health, the mathematical dependence of diabetes can be determined as follows (2):

$$K_{morb} = f(x_{age}, x_{GC}, x_{IC}, x_{DBP}, x_{BMI}, x_{gen}, x_{NP}, x_{ST}) \quad (2)$$

where  $x_{age}$  is the age of the patient;  $x_{GC}$  - glucose concentration after its oral glucose tolerance test;  $x_{IC}$  - insulin concentration;  $x_{DBP}$  - the value of the diastolic arterial test;  $x_{BMI}$  - value of body mass index;  $x_{gen}$  - genetic predisposition;  $x_{NP}$  - number of pregnancies;  $x_{ST}$  - the thickness of the skin fold in the triceps.

### THE PURPOSE AND OBJECTIVES OF THE STUDY

The object of study - the process of predicting and detecting diabetes by medical indicators of human health.

The subject of research - methods for predicting and classifying morbidity.

The aim of the work is to develop a modified genetic method for optimizing the parameters of the model based on the neural network to solve the problem of predicting the incidence of diabetes.

Research method - traditional models, neural networks, combined methods (neural networks and genetic algorithms, neural networks and multi-agent systems).

### DEVELOPMENT OF A MODIFIED GENETIC METHOD

A multilayer perceptron was chosen as a type of artificial neural network to build a model for predicting the risk of diabetes. The choice of this neural network is made on the basis of the problem to be solved, namely the classification of data to determine the risk of diabetes. The developed neural model has a dense structure in which each neuron is connected to all neurons of the next layer. The neural network consists of an input layer, two hidden layers, and one output neuron that defines the final class.

The outer layer has 8 neurons, according to the number of input parameters, the second layer of the network has 14 neurons, the third - 7 and the output layer has one neuron.

As a function of activation used sigmoidal function (3), which actually gives the value of the probability of disease risk:

$$f(x) = \frac{1}{1 + e^{-x}}, \quad (3)$$

For the initial initialization of synaptic scales, a normal distribution was used (mean value is 0, and standard deviation is 0.05) [14].

To solve this problem, it was decided to develop a modification of the GA, which will increase the accuracy of the neural network. (HA). It consists in parallel processing of a set of decisions, thus it focuses on the most

perspective of them that speaks of possibility of use of GA for the decision of a wide range of problems.

The solution of the problem is evaluated by calculating the values of the objective function. The type of objective function is determined by the nature of the problem for which the genetic algorithm is used. In this case, the objective function is taken as function (4), which calculates the number of correct responses of the neural network:

$$f = \sum_{i=1}^N \sum_{j=1}^m D(y_{ij} - y_{ij}^t), \quad (4)$$

where  $m$  is the number of outputs of the neural network;  $N$  - the number of samples in the training sample;  $y_{ij}$  - the actual value at the  $j$ -th output of the network for the image and;  $y_{ij}^t$  - the value that should be on the  $j$ -th output of the network for the image and;  $D$  is the value that depends on  $|y - y^t| < d$ , if the inequality is true, then  $D = 1$ , if not, then  $D = 0$ ;  $d$  is the value that determines the maximum allowable deviation of the real value from the target [15].

As a result of the calculation of the fitness function, it is determined whether the performance of GA will be continued. In the case of continued execution of GA, selection, crossing and mutation operations are performed sequentially.

A genetic algorithm can be used to adjust the weights of the neural network. To do this, you need to determine the target function (usually an error function) and the way the weights in the chromosome are represented. Each chromosome  $K = (k_1, k_2, \dots, k_i)$  Is a vector of genes or, in this case, weights (weights are read from the network in the prescribed order - from left to right and from top to bottom).

The length of the chromosome in accordance with the task of optimizing the neural network is calculated by formula (5):

$$p = N_1(L+1) + \sum_{\mu=1}^M N_{\mu}(N_{\mu-1} + 1), \quad (5)$$

where  $N_{\mu}$  - the number of neurons on the  $\mu$ -th layer;  $L$  - the number of features in the training sample;  $M$  - the number of layers of the neural network.

One of the problems that arise when using genetic methods is to determine not the global but the local extremum of the function being optimized. There are many modifications of the genetic method to solve this problem. In this project, it is proposed to choose parental couples based on their distant kinship.

In nature, there are two types of crossbreeding: related and unrelated. Analysis of numerous materials shows that the degree of kinship has a great influence on the offspring. If we take an example from nature, then



close kinship is accompanied by low viability of offspring, has an impact on its adaptability. When crossing unrelated individuals, such problems do not arise because harmful recessive mutations in the homozygous state will become heterozygous and will not affect the viability of the hybrid organism [16].

Thus, at the stage of choosing individuals for the formation of parental couples, it makes sense to take into account not only the rate of fitness of individuals, but also their relationship with each other so that couples are formed from the most distant individuals. As a result, the algorithm will scan unexplored areas and prevent convergence to the solution already found.

The degree of affinity can be expressed using formula (6):

$$d_{ij} = \sum_{k=1}^p |x_{ik} - x_{jk}|, \quad (6)$$

where  $x_{ik}$  - k-th gene in the chromosome of the i-th father;  $x_{jk}$  - k-th gene in the chromosome of the j-th father; p is the length of the chromosome.

This paper proposes two mutation operators, which differ from the classical one in that the mutation is carried out not only for a certain percentage of individuals, but also in a certain way the values of genes in chromosomes change. In the first proposed method, 20% of the population are subject to mutation. Next, two genes are randomly selected from the chromosome and the arithmetic mean of these values is calculated by formula (7):

$$H_A = \frac{h_{ij} + h_{ik}}{2}, \quad (7)$$

where  $h_{ij}$  - j-th gene of the i-th chromosome;  $h_{ik}$  - k-th gene of the i-th chromosome.

To prevent two genes with the same values from returning to the chromosome, it is proposed to change their values by adding and subtracting a certain number from a given range of values (8, 9).

$$h_{ij}^* = H_A + rand(0,1), \quad (8)$$

$$h_{ik}^* = H_A - rand(0,1), \quad (9)$$

where  $h_{ij}^*$ ,  $h_{ik}^*$  - value j-th and k-th gene of the i-th chromosome as a result of mutation; rand (0,1) is a random value from zero to one.

As a second modification of the mutation operator, it is proposed to calculate the arithmetic mean of two randomly selected genes (10), but unlike the first modification, in this one the value of one gene will be the value of the arithmetic mean, and the value of the second gene is zero (11):

$$h_{ij}^* = H_A + rand(0,1), \quad (10)$$

$$h_{ik}^* = 0, \quad (11)$$

where  $h_{ij}^*$ ,  $h_{ik}^*$  - value j-th and k-th gene of the i-th chromosome as a result of mutation; rand (0,1) is a random value from zero to one.

Thus, using the proposed operators, the meaning of the mutation operator is preserved, ie increases the diversity of the search and the introduction of new chromosomes into the population for a more complete study of the search space.

A multilayer perceptron was chosen as a type of artificial neural network to build a model for predicting the risk of diabetes. The choice of this neural network is made on the basis of the problem to be solved, namely the classification of data to determine the risk of diabetes.

After performing evolutionary operators, the population size is estimated, after which its size returns to baseline. The new population includes the most adapted individuals.

In addition, the optimization of the neural network by the particle swarm method was also implemented. Particle Swarm Optimization (PSO) is a stochastic optimization technique that takes the idea of the social behavior of a flock of birds or fish.

PSO has much in common with genetic algorithms [18]. The system is initialized by a population of random solutions and searches for optimums by updating the generations of these solutions. However, unlike genetic algorithms, PSO does not have such evolutionary operators as crossover and mutation. In PSO, potential solutions, called particles, fly through the solution space, focusing on the particles that are optimal at each time point.

At each point where the particle visited, the value of the objective function is calculated. In this case, each particle remembers what (and where) the best value of the objective function it personally found, and each particle knows where the point is located, which is the best among all the points that explored the particle. At each iteration, the particles adjust their speed (modulus and direction) to be closer to the best point that the particle found itself (the authors of the algorithm called this aspect of behavior "nostalgia"), and, at the same time, get closer to the point which is currently the best globally. After a number of iterations, the particles should gather near the best point, although it is possible that some of the particles will remain somewhere in a relatively good local extremum, but the main thing is that at least one particle is near the global extremum.

In the initial form of the algorithm, the speed correction was as follows (12):

$$v_{i,t+1} = v_{i,t} + \varphi_p r_p (p_i - x_{i,t}) + \varphi_g r_g (g_i - x_{i,t}), \quad (12)$$



where  $v_i$ ,  $t$  -  $i$ -th component of the speed at the  $t$ -th iteration of the algorithm;  $x_i$ ,  $t$  is the  $i$ -th coordinate of the particle at the  $t$ -th iteration of the algorithm;  $p_i$  is the  $i$ -th coordinate of the best solution found by the fraction;  $g_i$  is the  $i$ -th coordinate of the best solution found by all particles;  $r_p$  and  $r_g$  are random numbers in the interval (0, 1); and  $\omega$  - weights that need to be selected for a specific task.  $\varphi_p, \varphi_g$

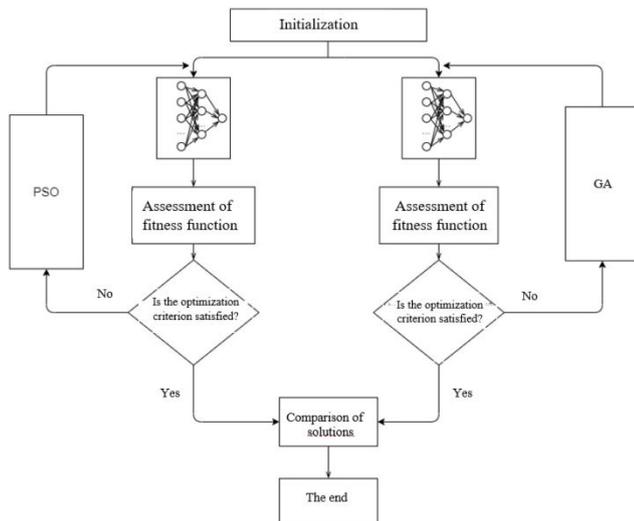
Then adjust the coordinate of each particle (13):

$$x_{i,t+1} = x_{i,t} + v_{i,t+1} \cdot \quad (13)$$

After that, the value of the objective function at each new point is calculated, and each particle checks whether the new coordinate has become the best among all the points it has visited. Then, among all new points, it is checked whether the best point is found globally and, if found, its coordinates and the value of the objective function at this point are remembered.

Compared to genetic algorithms, the advantages of PSO are ease of implementation and a small number of parameters that need to be set. PSOs are successfully used in many areas such as function optimization, training of artificial neural networks, fuzzy system control and others, where genetic algorithms can be used [15].

In general, the neural network optimization looks like shown in Figure-1.



**Figure-1.** Optimization of the neural network using GA and PSO.

## EXPERIMENTS IN TEACHING METHODS

Testing, as well as network training, was performed on a data set containing information on diabetes. The source of information about the disease is the National Institute of Diabetes, Digestive and Kidney Diseases (USA) [17]. The data set was formed from women 21 years and older of the Pima ethnic group. The set consists of 9 attributes and 768 instances:

a) the number of pregnancies;

- b) plasma glucose concentration 2 hours after administration in an oral glucose tolerance test;
- c) diastolic blood pressure (mm Hg);
- d) the thickness of the skin folds in the triceps (mm);
- e) the concentration of insulin in the serum ( $\mu\text{Ed} / \text{ml}$ );
- e) body mass index (weight in kg / (height in m)  $^2$ );
- g) a function that describes the genetic predisposition to diabetes (diabetes pedigree);
- h) age (years);
- i) a class that shows whether the patient had diabetes.

The established network was trained for 200 epochs. During testing, the method of dividing the data sample into training and test samples in the percentage of 80/20% was used.

The correctness of the classification is defined as the ratio between the correct answers and the total number of answers (14):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \cdot \quad (14)$$

where TP (True Positive) is the case when the real value was equal to 1 ( $x = 1$ ) and the answer of the algorithm is also 1 ( $y = 1$ ); TN (True Negative) - in a healthy patient the algorithm diagnosed the absence of disease; FP (False Positive) - the case when the real value was zero ( $x = 0$ ), and the predicted - one ( $y = 1$ ); FN (False Negative) - a case where the algorithm recognized a sick person healthy. In this case, since the problem of medical diagnosis is considered,  $y = 1$  means the presence of the disease, and  $y = 0$  - its absence.

It follows that ideally the model should give zero errors of classes False Positive and False Negative, but in fact this situation is almost unrealistic, so you should minimize the number of possible errors.

Binary crossentropy is used as a function of losses (15). Binary crossentropy calculates losses by calculating the following average:

$$loss = -\frac{1}{output\ size} \sum_{i=1}^{output\ size} y_i \cdot \log \hat{y}_i + (1 - y_i) \cdot \log(1 - \hat{y}_i), \quad (15)$$

where,  $\hat{y}_i$  -  $i$ -th scalar value at the input of the model;  $y_i$  - the corresponding target value; output size - the number of scalar data values at the output of the model.

It was used to estimate the ROC curve - a graph that illustrates the effectiveness of a binary classification system depending on a given class. The ROC curve provides detailed information about the behavior of the classifier. The curve is the result of the image on the True Positive rate (TPR) False Positive rate (FPR).

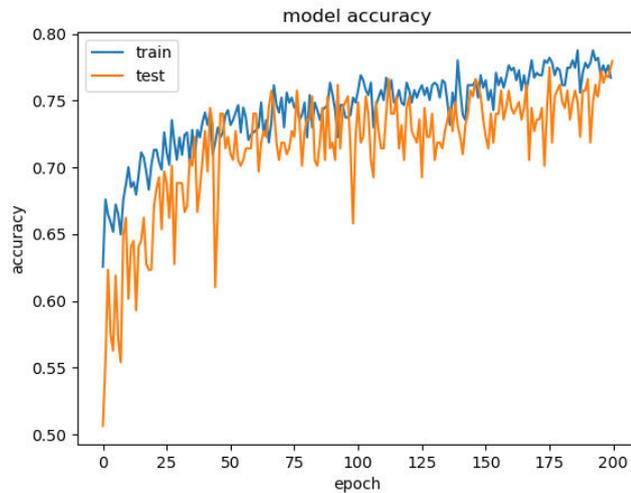
Another indicator that was used to evaluate and compare models is the running time of the model. The



time it takes to classify can be a sign of how fast the data can be processed.

Thus, testing of the developed software has the following results.

Ordinary fully connected perceptron showed the following results (Figures 2-4). The accuracy of diagnosis was 78.78%. graphical representation of the learning process of the model and its accuracy is shown in Figure-2 and in Table-1.



**Figure-2.** The accuracy of a fully connected perceptron.

From Figure-2 it can be seen that the accuracy of diagnosing this model was 78.78% and there is an almost constant value during the training of a fully connected perceptron.

The learning process of a fully connected perceptron according to the number of epochs is shown in Table-1, which shows the accuracy of a fully connected perceptron every 25 epochs of training in the training and test samples.

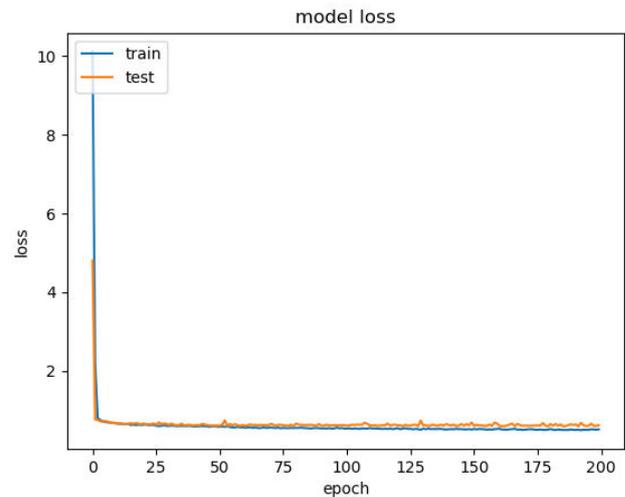
**Table-1.** Accuracy of a fully connected perceptron according to the epochs of learning.

Epoch	Accuracy in the training sample, %	Accuracy on the test sample, %
0	62.71	51.02
25	70.77	68.85
50	73.28	71.03
75	74.22	72.37
100	76.15	74.94
125	76.13	79.31
150	76.24	72.34
175	77.76	77.29
200	76.73	78.78

The change in the error value is shown in the graph (Figure-3), according to which there is a sharp

decrease in the error value, which shows a virtually constant value during the training of a fully connected perceptron.

The process of reducing the value of the loss function of a fully connected perceptron in the learning process according to the number of epochs is shown in Table-2, which shows the value of the loss function of a fully connected perceptron every 25 epochs of training in the training and test samples.



**Figure-3.** Error estimation of a fully connected perceptron.

**Table-2.** The value of the error function of a fully connected perceptron according to the epochs of learning.

Epoch	The value of the error function in the training sample	The value of the error function on the test sample
0	10.23	10.07
25	0.641	0.61
50	0.639	0.62
75	0.652	0.57
100	0.649	0.59
125	0.647	0.58
150	0.647	0.60
175	0.647	0.59
200	0.647	0.58

The value of the ROC curve is 0.811, the graph of which is shown in Figure-4 and in Table-3.

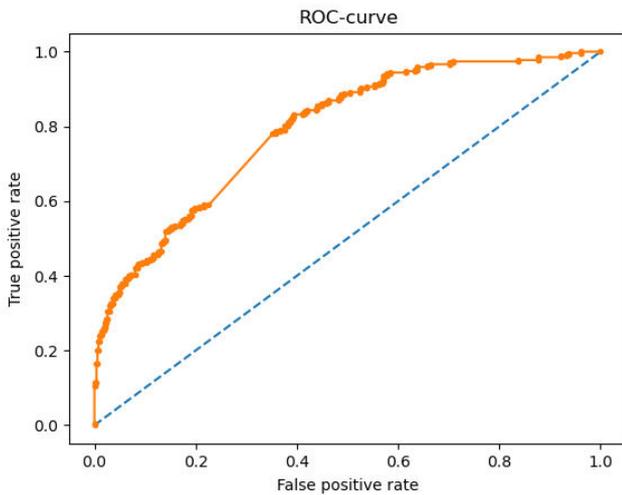


Figure-4. ROC-curve of a fully connected perceptron.

According to the increase in the value of False Positive rate, the change in the value of True Positive rate is shown in Table-3.

Table-3. Values of the ROC curve.

False Positive rate	0.0	0.2	0.4	0.6	0.8	1.0
True Positive rate	0.0	0.577 th most common	0.982	0.944	0.976 th most common	1.0

The results of a fully connected perceptron, the training of which was carried out using GA are shown in Figures 5-7.

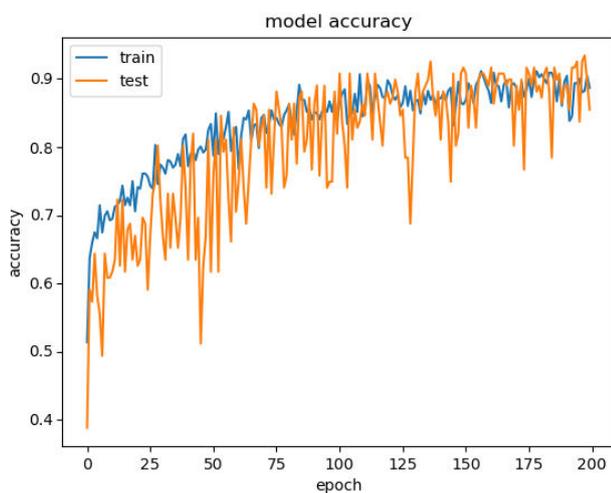


Figure-5. Accuracy of a fully connected perceptron studied with the help of GA.

From Figure-5 it can be seen that the accuracy of diagnosing this model was 89.3%.

The process of learning a fully connected perceptron using a genetic algorithm according to the

number of epochs is shown in Table-4, which shows the accuracy of a fully connected perceptron every 25 epochs of training in the training and test samples.

Table-4. Accuracy of a fully connected perceptron according to the epochs of learning.

Epoch	Accuracy in training sample, %	Accuracy on the test sample, %
0	51.93	38.71
25	74.24	67.86
50	80.47	77.12
75	84.17	87.78
100	87.21	90.74
125	87.34	84.34
150	87.56	91.02
175	89.06	89.98
200	88.87	89.3

The change in the value of the error is shown in the graph (Figure-6) according to which there is a decrease, but its value varies during the training of a fully connected perceptron in the range from 0.6 to 1.0.

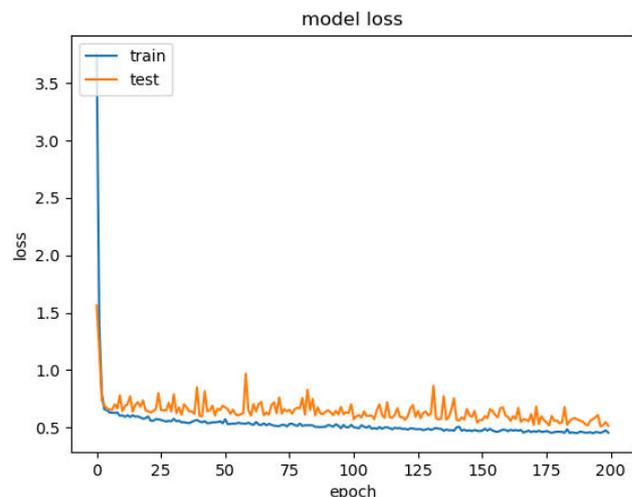


Figure-6. Estimation of the error of a fully connected perceptron studied with the help of GA.

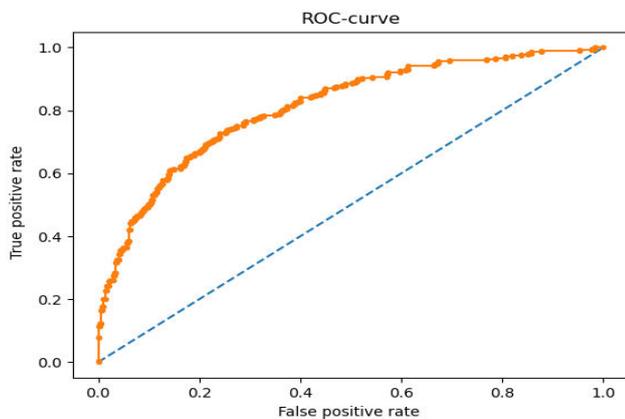
The process of reducing the value of the loss function of a fully connected perceptron in the process of learning the genetic algorithm in accordance with the number of epochs is shown in Table-5, which shows the value of the loss function of a fully connected perceptron every 25 epochs of training in the training and test samples.



**Table-5.** The value of the error function of a fully connected perceptron according to the epochs of learning.

Epoch	The value of the error function in the training sample	The value of the error function on the test sample
0	3,582 th most common	1,575 th most common
25	0.567 th most common	0.637
50	0.551 th most common	0.671 th most common
75	0.527 th most common	0.646
100	0.503	0.528
125	0.497 th most common	0.530
150	0.496	0.531 th most common
175	0.491	0.513 th most common
200	0.487	0.511

The graph of the ROC curve is shown in Figure-7 and in Table-6, the value of which is equal to 0.884.



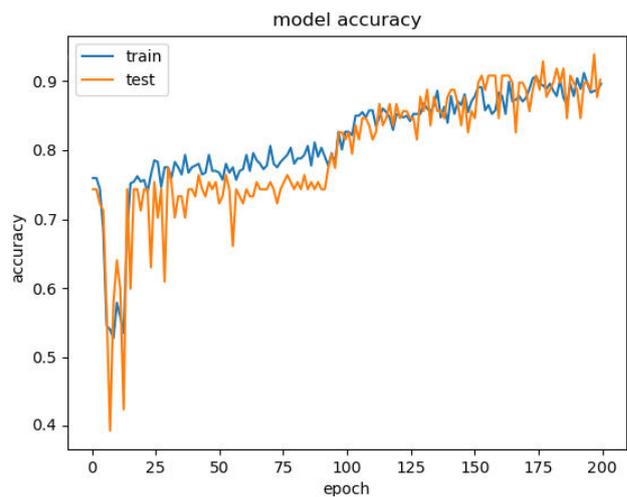
**Figure-7.** ROC-curve of a fully connected perceptron studied with the help of GA.

According to the increase in the value of False Positive rate, the change in the value of True Positive rate is shown in Table-6.

**Table-6.** Values of the ROC curve.

False Positive rate	0.0	0.2	0.4	0.6	0.8	1.0
True Positive rate	0.0	0.681	0.843 th most common	0.928	0.973	1.0

The results of a fully connected perceptron, which was trained using PSO are shown in Figures 8-10.



**Figure-8.** Accuracy of a fully connected perceptron studied with the help of PSO.

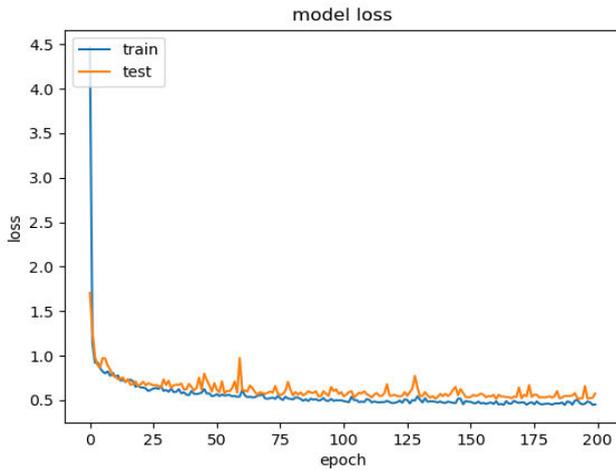
From Figure-8 it can be seen that the accuracy of diagnosing a fully connected perceptron, which was trained using PSO was 90.82%.

The process of learning a fully connected perceptron using the swarm of particles according to the number of epochs is shown in Table-7, which shows the accuracy of a fully connected perceptron every 25 epochs of training in the training and test samples.

**Table-7.** Accuracy of a fully connected perceptron according to the epochs of learning.

Epoch	Accuracy in the training sample, %	Accuracy on the test sample, %
0	76.23	74.49
25	78.35	73.05
50	76.57	73.31
75	78.59	75.47
100	82.78	82.38
125	84.76	84.53
150	87.88	89.55
175	90.43	89.27
200	89.94	90.82

The change in the value of the error is shown in the graph (Figure-9), according to which there is a decrease, but its value varies during network training in the range from 0.5 to 1.0.



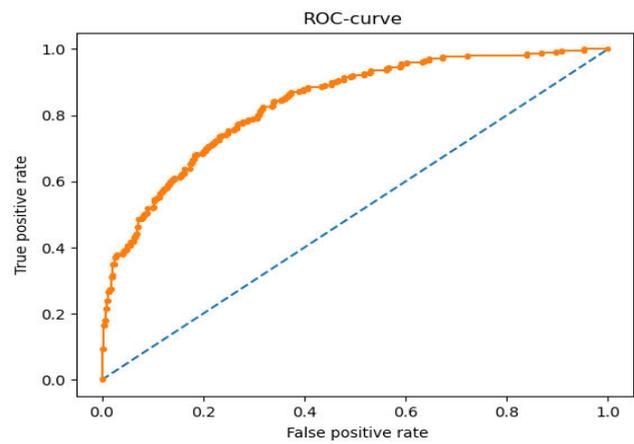
**Figure-9.** Estimation of PSO network learning error.

The process of reducing the value of the loss function of a fully connected perceptron, which teaches using the method of a swarm of particles in the learning process according to the number of epochs is shown in Table-8, which shows the value of the loss function of a fully connected perceptron every 25 epochs of training in the training and test samples.

**Table-8.** The value of the error function of a fully connected perceptron according to the epochs of learning.

Epoch	Accuracy in the training sample, %	Accuracy on the test sample, %
0	76.23	74.49
25	78.35	73.05
50	76.57	73.31
75	78.59	75.47
100	82.78	82.38
125	84.76	84.53
150	87.88	89.55
175	90.43	89.27
200	89.94	90.82

The graph of the ROC curve is shown in Figure-10 and in Table-9. Its value was 0.893.



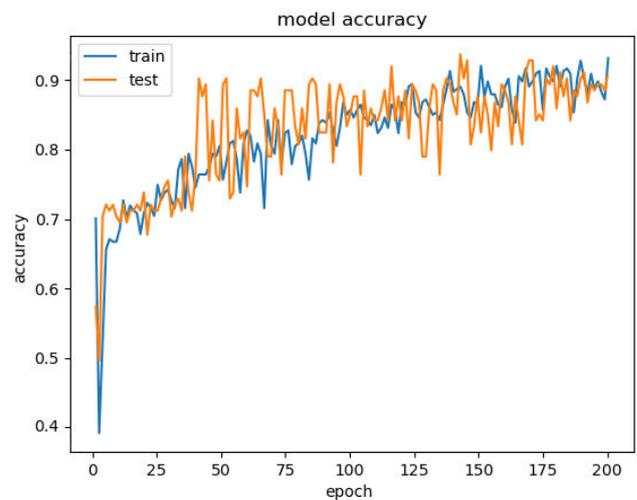
**Figure-10.** ROC-curve of a fully connected perceptron studied with PSO.

According to the increase in the value of False Positive rate, the change in the value of True Positive rate is shown in Table-9.

**Table-9.** The value of the ROC curve.

False Positive rate	0.0	0.2	0.4	0.6	0.8	1.0
True Positive rate	0.0	0.67	0.891 th most common	0.965	0.983	1.0

The results of a fully connected perceptron, the training of which was carried out using GA with modified selection operators and with the first of the proposed mutation operators are shown in Figures 11-13. the graph of accuracy of model is given in Figure-11 and in Table-10.



**Figure-11.** Accuracy of the neural network studied by means of the first modification of GA.

From Figure-11 it can be seen that the accuracy of diagnosing a fully connected perceptron, which was



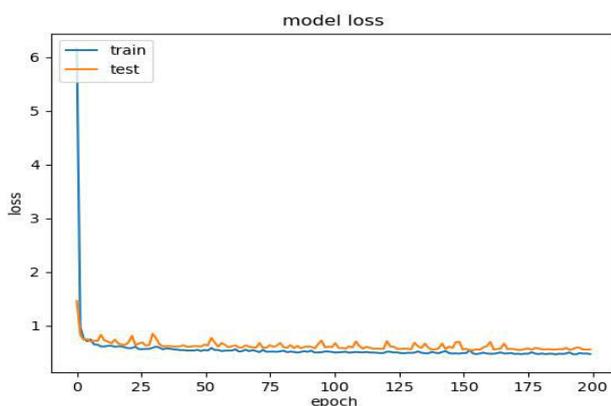
trained using GA with modified selection operators and with the first of the proposed mutation operators was 91.3%.

The process of learning a fully connected perceptron using a genetic algorithm with the proposed selection operator and the first approach to the mutation operator according to the number of epochs is shown in Table-10, which shows the accuracy of a fully connected perceptron every 25 epochs of training in the training and test samples.

**Table-10.** Accuracy of a fully connected perceptron according to the epochs of learning.

Epoch	Accuracy in the training sample, %	Accuracy on the test sample, %
0	70.58	57.78
25	73.85	71.23
50	77.56	84.79
75	82.53	88.62
100	85.37	85.10
125	85.41	88.74
150	88.90	85.98
175	89.12	85.37
200	83.26	91.3

The change in the error value is shown in the graph (Figure-12), according to which there is a decrease in the value with minor fluctuations in the range of 0.5-0.75.



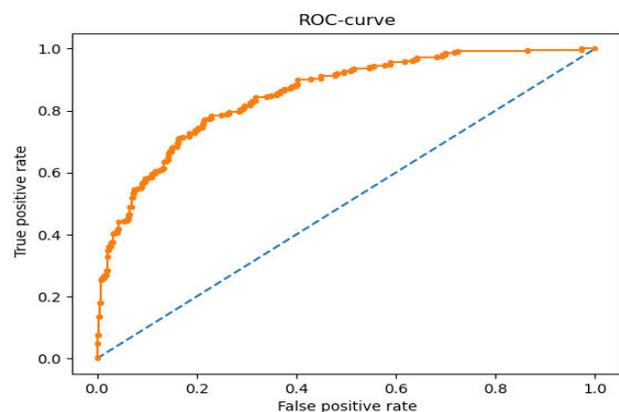
**Figure-12.** Estimation of the error of a fully connected perceptron studied with the help of the first modification of GA.

**Table-11.** The value of the error function of a fully connected perceptron according to the epochs of learning.

Epoch	The value of the error function in the training sample	The value of the error function on the test sample
0	6,975 th most common	1.50
25	0.563 th most common	0.679
50	0.471 th most common	0.619
75	0.468	0.617
100	0.403	0.615
125	0.407	0.613
150	0.410	0.612
175	0.406	0.611
200	0.404	0.610

The process of reducing the value of the loss function of a fully connected perceptron, which trains using a genetic algorithm with the proposed selection operator and the first approach to the mutation operator in the learning process according to the number of epochs is shown in Table-11, which shows the value of the loss function of a fully connected perceptron every 25 epochs of training in the training and test samples.

The value of the ROC curve is 0.9, the graph of which is shown in Figure-13 and in Table-12.



**Figure-13.** ROC-curve of a fully connected perceptron, studied with the first modification of GA.

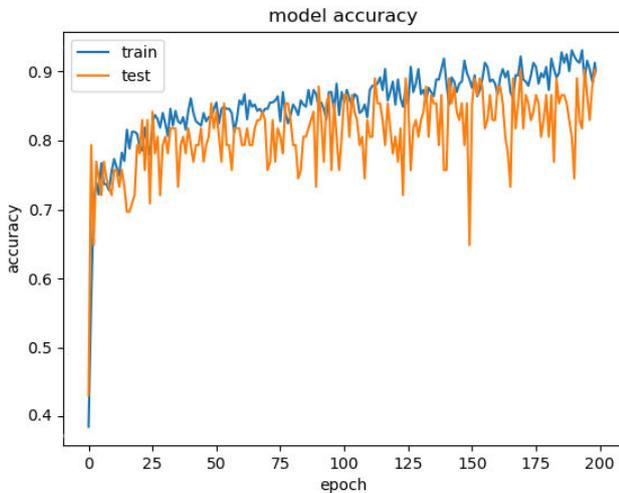
According to the increase in the value of False Positive rate, the change in the value of True Positive rate is shown in Table-12.

**Table-12.** The value of the ROC curve.

False Positive rate	0.0	0.2	0.4	0.6	0.8	1.0
True Positive rate	0.0	0.689	0.896 th most common	0.961	0.985	1.0



The results of a fully connected perceptron, the training of which was carried out using GA with modified selection operators and with the second of the proposed mutation operators are shown in Figures 14-16.



**Figure-14.** Accuracy of a fully connected perceptron, studied with the help of the second modification of GA.

From Figure-14 it can be seen that the accuracy of diagnosing a fully connected perceptron, which was trained using GA with modified selection operators and with the second of the proposed mutation operators was 89.97%.

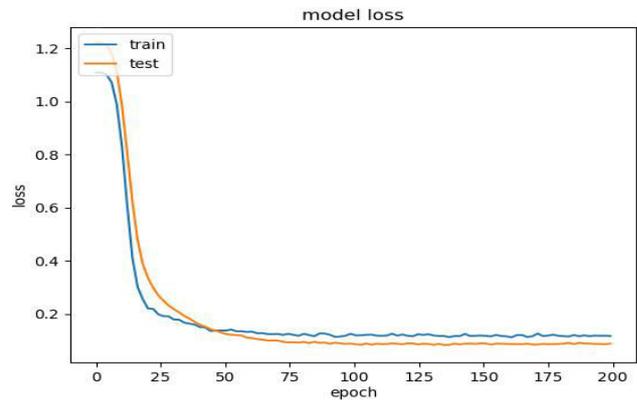
The process of learning a fully connected perceptron using a genetic algorithm with the proposed selection operator and the second approach to the mutation operator according to the number of epochs is shown in Table-13, which shows the accuracy of a fully connected perceptron every 25 epochs of training in the training and test samples.

**Table-13.** Accuracy of a fully connected perceptron according to the epochs of learning.

Epoch	Accuracy in training sample, %	Accuracy on the test sample, %
0	36.74	43.31
25	82.94	83.74
50	84.52	82.76
75	82.98	80.73
100	86.69	86.65
125	87.43	82.83
150	87.85	85.56
175	90.38	86.67
200	89.14	89.97

The change in the error value is shown in the graph (Figure-15) and in Table-14, according to which

there is a gradual decrease in the value of the error, which maintains a stable value since the 75th era.



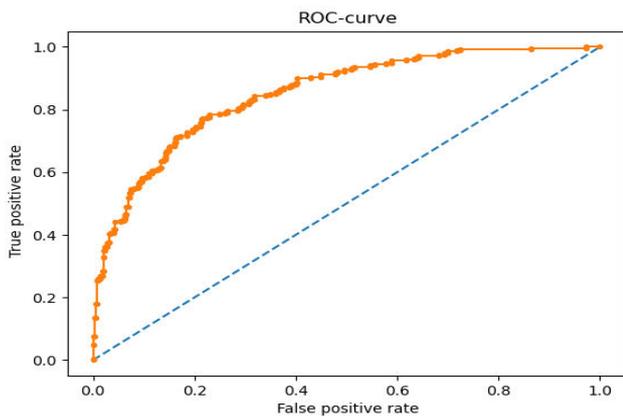
**Figure-15.** Estimation of the error of a fully connected perceptron studied with the help of the second modification of GA.

**Table-14.** The value of the error function of a fully connected perceptron according to the epochs of learning.

Epoch	The value of the error function in the training sample	The value of the error function on the test sample
0	1,157 th most common	1.33
25	0.198 th most common	0.264 th most common
50	0.146 th most common	0.132
75	0.125	0.088
100	0.123	0.081
125	0.124	0.080
150	0.123	0.081
175	0.124	0.080
200	0.123	0.080

The process of reducing the value of the loss function of a fully connected perceptron, that training using a genetic algorithm with the proposed selection operator and the first approach to the mutation operator in the learning process according to the number of epochs is shown in Table-14, which shows the value of the loss function of a fully connected perceptron every 25 epochs of training in the training and test samples.

The value of the ROC curve is equal to 0.899. The graph of this curve is shown in Figure-16 and in Table-15.



**Figure-16.** ROC-curve of a fully connected perceptron, studied using the second modification of GA.

According to the increase in the value of False Positive rate, the change in the value of True Positive rate is shown in Table-15.

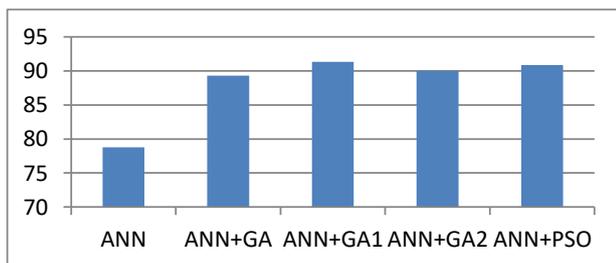
**Table-15.** The value of the ROC curve.

False Positive rate	0.0	0.2	0.4	0.6	0.8	1.0
True Positive rate	0.0	0.743 th most common	0.872 th most common	0.957	0.983	1.0

To visualize the results obtained during the study of the models, the data are listed in table. 16, and the accuracy of the classification is reflected in the histogram (Figure-17).

**Table-16.** Comparative table of model evaluation results.

No	Name	Accuracy, %	The value of the ROC curve	Time, s
1	Neural network	78.78	0.811	117
2	Neural network + HA	89.3	0.884	99
3	Neural networks + the first modification of GA	91.3	0.9	101
4	Neural networks + the second modification of GA	89.97	0.899 th most common	102
5	Neural network + PSO	90.82	0.893	97



**Figure-17.** Histogram comparing the accuracy of the tested models.

The results of comparing the values of the accuracy of all experiments are shown in the histogram (Figure-17). It is seen that learning neural networks using evolutionary methods significantly increases their accuracy as a classifier.

**DISCUSSION**

From the results of the experiments we can conclude that the training of the network using evolutionary algorithms shows a significant improvement compared to standard methods. The use of these methods shows approximately the same result in terms of diagnostic accuracy, and the time spent on the work is quite small.

From Figure-26 shows that the best accuracy was shown by a fully connected perceptron trained using a genetic algorithm with the proposed selection operator and the first approach to the mutation operator, the accuracy of which was 91.3%. A fully connected perceptron studied using the particle swarm method showed an accuracy of 90.82%. Slightly worse results are obtained by a fully connected perceptron, which was studied using a genetic algorithm with the proposed selection operator and the second approach to the mutation operator, the accuracy of which is 89.97%. A fully connected perceptron trained using a genetic algorithm showed an accuracy of 89.3%. The worst result in terms of accuracy has a fully connected perceptron, studied using the gradient method, its accuracy was 78.78%.

**CONCLUSIONS**

During the work, the classical methods that can be applied to the classification problem were analyzed, namely the Bayesian network, NNge, Random Forest, HMM, RBFNetwork, AdaBoost and ENORA. It is determined that they do not give satisfactory accuracy in data classification. The Bayesian network and the Random Forest algorithm show themselves in the best way. Also for comparison, a multilayer perceptron, a convolutional neural network, an LSTM network, and a fully connected NM were analyzed. The LSTM network showed the best



accuracy, but it has a rather complex structure for the classification task.

As a result of the work, a method of learning the neural network using a genetic algorithm is proposed, for which a method of selecting individuals for parental pairs based on their relatedness and two methods of mutation of chromosome genes is proposed. Two approaches to the mutation operator are proposed. When muting, the values of the selected genes are calculated based on their average values and then added, subtracted or reset the value.

During the creation and training of a model based on a network of long short-term memory, the possibility of using the method of particle swarm and genetic algorithm to adjust the weights of the network was investigated. According to the results of testing and experimental research, it is determined that evolutionary methods significantly increase the accuracy of classification. The accuracy of classification of a fully connected perceptron was 78.78%. The accuracy of classification of a fully connected perceptron studied using a genetic algorithm was 89.3%, using PSO - 90.82%. A fully connected perceptron trained using a genetic algorithm with the proposed selection operators and the first approach to mutation showed an accuracy of 91.3%, and with the second approach to mutation the accuracy of the classifier was 89.97%. According to the results of studies of the proposed genetic algorithm, we can say that there is an improvement in the accuracy of classification, namely in the first case compared with classical HA by 2%, in the second case by 0.67%. The obtained results allow to offer effective methods for improving the quality of predicting the presence of diabetes, which will improve the process of diagnosis and treatment of the disease.

#### ACKNOWLEDGMENT

The authors would like to thank Centre for Research and Innovation Management (CRIM) for the support given to this research by Universiti Teknikal Malaysia Melaka (UTeM). We thank also those who contributed in any other forms for providing their continuous support throughout this work.

#### REFERENCES

- [1] Parte RS, Patil A., Patil A., Kad A., Kharat S. 2019. Non-invasive method for diabetes detection using CNN and SVM classifier. *International journal of research in engineering, science and management*. 2: 659-661. ISSN: 2581-7175.
- [2] Jahani M., Mahdavi M. 2016. Comparison of Predictive Models for the Early Diagnosis of Diabetes. *Healthcare Informatics Research*. 22: 95-100. doi: 10.4258 / hir.2016.22.2.95.
- [3] Farre P., Heurteau A., Cuvier O., Emberly E. 2018. Dense neural networks for predicting chromatin conformation. *BMC Bioinformatics*. Vol. 19. doi: 10.1186 / s12859-018-2286-z.
- [4] Lekha S., Suchetha M. 2018. A Novel 1-D convolution neural network with SVM architecture for real-time detection applications. *IEEE Sensors Journal*. 18: 724-731. doi: 10.1109 / jsen.2017.2780178.
- [5] Lekha S. 2018. MS Real-Time Non-Invasive Detection and Classification of Diabetes Using Modified Convolution Neural Network. *IEEE Journal of Biomedical and Health Informatics*. 22: 1630-1636. doi: 10.1109 / jbhi.2017.2757510.
- [6] Balaj H., Iyengar N., Caytiles R. 2017. Optimal Predictive analytics of Pima Diabetics using Deep Learning. *International Journal of Database Theory and Application*. 10: 47-62. doi: 10.14257 / ijtda.2017.10.9.05.
- [7] Lai, H., Huang, H., Keshavjee, K., Guergachi, A., Gao, X. 2019. Predictive models for diabetes mellitus using machine learning techniques. *BMC Endocrine Disorders*, vol. 19. doi: 10.1186 / s12902-019-0436-6.
- [8] Nimmagadda, SP, Yeruva, S., Siempu, R. 2019. Improved diabetes prediction model for predicting Type-II diabetes. *International journal of innovative technology and exploring engineering*, vol. 8, pp. 230-235. doi: 10.35940 / ijitee.13594.1081219.
- [9] Ali, R., Hussain, J., Siddiqi, M., Hussain, M., Lee, S. 2015. H2RM: A Hybrid Rough Set Reasoning Model for Prediction and Management of Diabetes Mellitus. *Sensors*, vol. 15, pp. 15921-15951. doi: 10.3390 / s150715921.
- [10] Pavate, A., Ansari, N. 2015. Risk Prediction of Disease Complications in Type 2 Diabetes Patients Using Soft Computing Techniques. 2015 Fifth International Conference on Advances in Computing and Communications (ICACC). doi: 10.1109 / icacc.2015.61.
- [11] Li, G., Watanabe, K., Anzai, H., Song, X., Qiao, A., Ohta, M. 2019. Pulse-Wave-Pattern Classification with a Convolutional Neural Network. *Scientific Reports*, vol. 9. Doi: 10.1038 / s41598-019-51334-2.
- [12] Jiang H., You Z., Huang Y. 2019. Predicting drug-disease associations via sigmoid kernel-based convolutional neural networks. *Journal of*



- Translational Medicine. Vol. 17. doi: 10.1186 / s12967-019-2127-5.
- [13] Al-Azzawi D. 2019. Human Age and Gender Prediction Using Deep Multi-Task Convolutional Neural Network. Journal of Southwest Jiaotong University. Vol. 54. doi: 10.35741 / issn.0258-2724.54.4.11.
- [14] Hemanth D., Anitha J., Naaji A., Geman O., Popescu D., Hoang Son L. 2019. A Modified Deep Convolutional Neural Network for Abnormal Brain Image Classification. IEEE Access. 7: 4275-4283. doi: 10.1109 / access.2018.2885639.
- [15] Panda M. 2018. Combining multi-objective evolutionary algorithm with averaged one-dependence estimators for big data analytics. International Journal of Computational Intelligence Studies. 7: 1. doi: 10.1504/ijcistudies.2018.090160.
- [16] Sisodia D., Sisodia D. 2018. Prediction of Diabetes using Classification Algorithms. Computer Science Procedure. 132: 1578-1585. doi: 10.1016 / j.procs.2018.05.122.
- [17] Zou Q., Qu K., Luo Y., Yin D., Ju Y., Tang H. 2018. Predicting Diabetes Mellitus With Machine Learning Techniques. Frontiers in Genetics. Vol. 9. doi: 10.3389 / fgene.2018.00515.
- [18] Kalyankar G., Poojara S., Dharwadkar N. 2017 Predictive analysis of diabetic patient data using machine learning and Hadoop. International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC). doi: 10.1109 / i-smac.2017.8058253.
- [19] Wang F., *et al.* 2016. Application of genetic algorithm-back propagation for prediction of mercury speciation in combustion flue gas. Clean Technologies and Environmental Policy. 18(4): 1211-1218. doi: 10.1007 / s10098-016-1095-1.
- [20] Kong Z., *et al.* 2020. Error prediction and structure determination for CMAC neural network based on the uniform design method. Expert Systems. 38(2). DOI: 10.1111 / exsys.12614.
- [21] Stepanenko A., Oliinyk A., Fedorchenko I. 2020. Application of the fuzzy clustering technique for processing and analysis of medical images. CEUR Workshop Proceedings. 3rd International Workshop on Computer Modeling and Intelligent Systems, CMIS 2020; National University "Zaporizhzhia Polytechnic" Zaporizhzhia; Ukraine; 27 April 2020 до 1 May 2020, pp. 1079-1093. ISSN: 16130073.
- [22] Li Q., Wu J., Wei H. 2018. Reduction of elemental mercury in coal-fired boiler flue gas with computational intelligence approach. Energy. pp. 753-762. doi: 10.1016 / j.energy.2018.07.037.
- [23] Oliinyk A., Fedorchenko I., Stepanenko A., Rud M., Goncharenko D. 2018. Evolutionary method for solving the traveling salesman problem. 2018 5th International Scientific-Practical Conference (PICST), Kharkiv: Kharkiv National University of Radioelectronics. pp. 331 - 339. DOI: <https://doi.org/10.1109/INFOCOMMST.2018.8632033>.
- [24] Oliinyk A., Fedorchenko I., Zaiko T., Goncharenko D., Stepanenko A., Kharchenko A. 2019. Development of genetic methods of network pharmacy financial indicators optimization. 2019 IEEE International Scientific-Practical Conference: Problems of Infocommunications Science and Technology, PIC S and T 2019; Kyiv; Ukraine; 8 October 2019 до 11 October 2019, pp. 607-612, 2018. DOI: <https://doi.org/10.1109/PICST47496.2019.9061396>.
- [25] Oliinyk A., Fedorchenko I., Stepanenko A., Rud M., Goncharenko D. 2019. Combinatorial optimization problems solving based on evolutionary approach. 2019 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM). pp. 41-45, 2019. DOI: 10.1109 / CADSM.2019.8779290.
- [26] Fedorchenko I., Oliinyk A., Stepanenko O., Zaiko T., Svyrydenko A., Goncharenko D. 2019. Genetic method of image processing for motor vehicle recognition 2nd International Workshop on Computer Modeling and Intelligent Systems, CMIS 2019; CEUR Workshop Proceedings. Zaporizhzhia; Ukraine; 15 April 2019 до 19 April 2019, vol. 2353, pp. 211-226. ISSN: 16130073
- [27] Oliinyk A., Fedorchenko I., Stepanenko A., Katschan A., Fedorchenko Y., Kharchenko A., Goncharenko D. 2019. Development of genetic methods for predicting the incidence of volumes of pollutant emissions in air. 2<sup>nd</sup> International Workshop on Informatics and Data-Driven Medicine. pp. 340-353. ISSN: 16130073.



- [28] Fedorchenko I., Oliinyk A., Stepanenko A., Zaiko T., Shylo S., Svyrydenko A. 2019. Development of the modified methods to train a neural network to solve the task on recognition of road users. Eastern European Journal of Enterprise Technologies, issue 9/98, pp. 46-55. DOI: 10.15587 / 1729-4061.2019.164789.
- [29] Fedorchenko I., Oliinyk A., Stepanenko A., Zaiko T., Korniienko S., Burtsev N. 2019. Development of a genetic algorithm for placing power supply sources in a distributed electric network. Eastern European Journal of Enterprise Technologies. 5/3(101): 6-16, 2019. doi: 10.15587 / 1729-4061.2019.180897.
- [30] Fedorchenko I., Oliinyk A., Stepanenko O., Kharchenko A. 2020. Development of the genetic algorithm for technical diagnostics of aircraft. Information Technology and Interactions (IT&I-2020), CEUR Workshop Proceedings. 2833: 11-25, ISSN: 16130073.
- [31] Fedorchenko I., Oliinyk A., Stepanenko A., Zaiko T., Korniienko S., Kharchenko A. 2020. Construction of a genetic method to forecast the population health indicators based on neural network models. Eastern-European Journal of Enterprise Technologies. 1(4103): 52-63, DOI: 10.15587 / 1729-4061.2020.197319.
- [32] Oliinyk A., Fedorchenko I., Donenko V., Stepanenko A., Korniienko S., Kharchenko A. 2020. Development of an evolutionary optimization method for financial indicators of pharmacies. International Journal of Computing. 19(3): 449-463, 2020. ISSN 1727-6209
- [33] Goncharenko D., Oliinyk A., Fedorchenko I., Korniienko S., Stepanenko O.O., Kharchenko A.S., Fedorchenko Y. 2020. Genetic Algorithm for Solution of the Problem of Optimal Location of the Distributed Electrical Networks. 10<sup>th</sup> International Conference on Advanced Computer Information Technologies (ACIT), Deggendorf, Germany. pp. 380-385. Doi: 10.1109/ACIT49673.2020.9208888.
- [34] Oliinyk A., Fedorchenko I., Stepanenko A., Rud M., Goncharenko D. 2021. Implementation of Evolutionary Methods of Solving the Travelling Salesman Problem in a Robotic Warehouse, Lecture Notes on Data Engineering and Communications Technologies. 48: 263-292. doi: 10.1007/978-3-030-43070-2\_13.
- [35] Oliinyk A., Subbotin A. 2015. The decision tree construction based on a stochastic search for the neuro-fuzzy network synthesis. Optical Memory and Neural Networks (Information Optics). 24(1): 18-27. DOI: 10.3103/S1060992X15010038
- [36] Oliinyk A., Skrupsky S., Subbotin S. 2017. Parallel Computer System Resource Planning for Synthesis of Neuro-Fuzzy Networks. Recent Advances in Systems, Control and Information Technology. Advances in Intelligent Systems and Computing. 543: 88-96. DOI: 10.1007/978-3-319-48923-0\_12
- [37] Oliinyk A. O., Skrupsky S., Subbotin S., Korobiichuk I. 2017. Parallel Method of Production Rules Extraction Based on Computational Intelligence. Automatic Control and Computer Sciences. 51(4): 215-223. DOI: 10.3103/S0146411617040058
- [38] Oliinyk A. O., Skrupsky S., Subbotin S. 2018. Experimental research and analysis of complexity of parallel method for production rules extraction. Automatic Control and Computer Sciences. 52(2): 89-99. DOI: 10.3103/S0146411618020062
- [39] Oliinyk A., Subbotin S., Lovkin V., Leoshchenko S., Zaiko T. Feature selection based on parallel stochastic computing. Computer Science and Information Technologies, (CSIT'2018) : XIII International Scientific and Technical Conference, Lviv, 11-14 September, 2018 : proceedings. – Lviv: Vegha i Ko, pp. 347–351. DOI: 10.1109/STC-CSIT.2018.8526729
- [40] Alsayaydeh J.A.J., Aziz A., Rahman A.I.A., Abbasi M.I., Khang A.W.Y. 2021. Development of Programmable Home Security using GSM System for Early Prevention. ARPN Journal of Engineering and Applied Sciences. 16(1): 88-97.
- [41] Alsayaydeh J.A.J., Khang W.A.Y., Hossain A.K.M.Z, Shkarupylo V., Puspanathan J. 2020. The experimental studies of the automatic control methods of magnetic separators performance by magnetic product. ARPN Journal of Engineering and Applied Sciences. 15(7): 922-927.
- [42] Alsayaydeh J.A.J., Indra W.A., Khang W.A.Y., Shkarupylo V., Jkatisan D.A.P.P. 2019. Development of vehicle ignition using fingerprint. ARPN Journal of Engineering and Applied Sciences. 14(23): 4045-4053.
- [43] Alsayaydeh J.A.J., Khang W.A.Y., Indra W.A., Puspanathan J.B., Shkarupylo V., Zakir Hossain



A.K.M., Saravanan S. 2019. Development of vehicle door security using smart tag and fingerprint system. ARPJN Journal of Engineering and Applied Sciences. 9(1): 3108-3114.

[44] Alsayaydeh J.A.J., Khang W.A.Y., Indra W.A., Shkarupylo V., Jayasundar J. 2019. Development of smart dustbin by using apps. ARPJN Journal of Engineering and Applied Sciences. 14(21): 3703-3711.

[45] Alsayaydeh J.A., Shkarupylo V., Hamid M. S., Skrupsky S., Oliinyk. A. 2018. Stratified Model of the Internet of Things Infrastructure. Journal of Engineering and Applied Sciences. 13(20): 8634-8638, 2018. doi: 10.3923/jeasci.2018.8634.8638

[46] Alsayaydeh J.A., Nj M., Syed S.N., Yoon A.W., Indra W.A., Shkarupylo V., Pellipus C. 2019. Homes appliances control using Bluetooth. ARPJN Journal of Engineering and Applied Sciences. 14(19): 3344-3357