Computers in Biology and Medicine 134 (2021) 104431

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

Comparison of various approaches to combine logistic regression with genetic algorithms in survival prediction of hepatocellular carcinoma

Wojciech Książek [a], Michał Gandor [a], Paweł Pławiak [a,b,*]

[a] Department of Computer Science, Faculty of Computer Science and Telecommunications, Cracow University of Technology, Krakow, Poland
[b] Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Gliwice, Poland

ARTICLE INFO

Keywords:
Logistic regression
Machine learning
Hepatocellular carcinoma
Genetic algorithms

ABSTRACT

Hepatocellular carcinoma (HCC) is the most common liver cancer in adults. Many different factors make it difficult to diagnose in humans.. In this paper, a novel diagnostics approach based on machine learning techniques is presented. Logistic regression is one of the most classic machine learning models used to solve the problem of binary classification. In typical implementations, logistic regression coefficients are optimized using iterative methods. Additionally, parameters such as solver, C - a regularization parameter or the number of iterations of the algorithm operation should be selected. In our research, we propose a combination of logistic regression with genetic algorithms. We present three experiments showing the fusion of those methods. In the first experiment, we genetically select the logistic regression parameters, while the second experiment extends this approach by including a genetic selection of features. The third experiment presents a novel approach to train the logistic regression model - the genetic selection of coefficients (weights). Our models are tested for the survival prediction of hepatocellular carcinoma based on patient data collected at Coimbra's Hospital and Universitary Center (CHUC), Portugal. The model we proposed achieved a classification accuracy of 94.55% and an f1-score of 93.56%. Our algorithm shows that machine learning techniques optimized by the proposed concept can bring a new and accurate approach in HCC diagnosis with high accuracy.

1. Introduction

In recent years, we are witnessing the constant growth of the amount of data we store. This creates new challenges which besides big data storage, also includes its interpretation. Machine learning algorithms make it possible to interpet the data in a specific manner that humans cannot handle. These algorithms are characterized by their efficiency, flexibility and ability to generalize the data they process. That is one of the reasons why ML enabled the engineers to fulfil the demands of today's world including the analysis of credit scoring, advertisements and much faster treatment of complicated illnesses. Machine learning algorithms are successfully used in ECG interpretations including abnormal heart rate diagnoses [18], arrhythmia detection [38] or assessing of electrocardiogram visual interpretation strategies [34]. In medicine ML is also used to diagnose coronary artery diseases [1,2,30,49], breast cancers [3], wart diseases [4], heartbeat classifications [24] and Alzheimer's disease [6]. On the other hand, ML can also be used in other fields like credit scoring [36], approximating of phenol concentration [39], modelling of the results of tympanoplasty in chronic suppurative otitis media patients [45], chemical analysis [46], assessing of dots and globules in dermoscopic colour images [23]. In this paper, we also use genetic algorithms to accelerate the computation and get more precise results [37].

Hepatocellular carcinoma (HCC) is one of the leading causes of death associated with cancer-related deaths worldwide [11]. One of the most promising prevention approaches is an early diagnosis [16], but the number of different factors causing cancer makes it hard to distinguish it from other diseases, especially at an early stage. We believe that machine learning methods can be widely used for various types of problems making the analysis faster and less error prone.

There are two main reasons that drove us to apply machine learning in the HCC problem. First, HCC is the leading type of liver cancer worldwide. There are many factors that can indicate the presence of cancer. Accurate and early diagnosis can prevent many deaths and improve life quality. The second reason is that using genetic algorithms, the accuracy of the prediction models can be improved compared to

* Corresponding author. Department of Computer Science, Faculty of Computer Science and Telecommunications, Cracow University of Technology, Krakow, Poland.
E-mail addresses: plawiak@pk.edu.pl, plawiak@iitis.pl (P. Pławiak).

https://doi.org/10.1016/j.compbiomed.2021.104431
Received 13 January 2021; Received in revised form 14 April 2021; Accepted 21 April 2021
Available online 11 May 2021

solutions already present in literature.

The main goal of this study is to design a novel logistic regression learning algorithm using genetic algorithms. The algorithm implemented is based on 49 different features, which are considered to be key factors causing HCC. It allows for faster diagnosis taking into consideration all of the provided indicators simultaneously.

The main contributions to our work are: (1) investigation and preprocessing of the HCC dataset, (2) examination of the standard logistic regression learning, (3) applying GA to the logistic regression learning, (4) comparison of the results.

This study introduces a novel method of determining logistics regression coefficients. The computation is done using genetic algorithms through data classification provided in the Hepatocellular Carcinoma dataset [42]. The aim of the experiment is to efficiently train the logistic regression model and classify the data into two classes present in the dataset - die or live. The experiment includes a deep analysis of preprocessing paths, including missing values replacement and scaling methods which are implemented within the experiment. Various approaches to combine logistic regressions are presented in this study.

The paper is organized as follows: after the introduction of the examined problem, we deeply describe the methods used in section 2. Later on, we give a brief introduction of logistic regression as the main algorithm used in the analysis 3. The next section 4 is fully aimed at describing the performed experiments, including genetic algorithms applied in different paths we took. The two last sections include detailed results - comparison 5 and discussion 6.

1.1. HCC dataset

The dataset was collected at Coimbra's Hospital and University Centre (CHUC), Portugal. It contains data on 165 patients described by 49 features [41]. There are 23 quantitative variables and 26 qualitative variables. Lots of missing values are present (10.22%). Moreover, only eight patients have complete information in all fields (4.85%). The dataset has unbalanced classes (63 vs 102).

Using the random forest algorithm (with 1000 estimators), we analysed the significance of the data set features [28]. According to the algorithms, the most important features are the following: International

Table 1
Details of HCC dataset [41].

#	Feature	Range	Type	Missing	Mean	Std
1	Gender	0, 1	binary	0		
2	Symptoms	0, 1	binary	18		
3	Alcohol	0, 1	binary	0		
4	Hepatitis B Surface Antigen	0, 1	binary	17		
5	Hepatitis B e Antigen	0, 1	binary	39		
6	Hepatitis B Core Antibody	0, 1	binary	24		
7	Hepatitis C Virus Antibody	0, 1	binary	9		
8	Cirrhosis	0, 1	binary	0		
9	Endemic Countries	0, 1	binary	39		
10	Smoking	0, 1	binary	41		
11	Diabetes	0, 1	binary	3		
12	Obesity	0, 1	binary	10		
13	Hemochromatosis	0, 1	binary	23		
14	Arterial Hypertension	0, 1	binary	3		
15	Chronic Renal Insufficiency	0, 1	binary	2		
16	Human Immunodeficiency Virus	0, 1	binary	14		
17	Nonalcoholic Steatohepatitis	0, 1	binary	22		
18	Esophageal Varices	0, 1	binary	52		
19	Splenomegaly	0, 1	binary	15		
20	Portal Hypertension	0, 1	binary	11		
21	Portal Vein Thrombosis	0, 1	binary	3		
22	Liver Metastasis	0, 1	binary	4		
23	Radiological Hallmark	0, 1	binary	2		
24	Age at diagnosis	20–93	scale	0	64.7	13.37
25	Grams of Alcohol per day	0–500	scale	48	71.01	76.28
26	Packs of cigarets per year	0–510	scale	53	20.46	51.57
27	Performance Status	0, 1, 2, 3, 4	ordinal	0	1.02	1.18
28	Encefalopathy degree	0, 1, 2, 3	ordinal	1	1.16	0.43
29	Ascites degree	0, 1, 2, 3	ordinal	2	1.44	0.69
30	International Normalised Ratio	0.84–4.82	scale	4	1.42	0.48
31	Alpha-Fetoprotein (ng/mL)	1.2–1810348	scale	8	19299.95	149098.34
32	Haemoglobin (g/dL)	5–18.7	scale	3	12.88	2.15
33	Mean Corpuscular Volume (fl)	69.5–119.6	scale	3	95.12	8.41
34	Leukocytes(G/L)	2.2–13000	scale	3	1473.96	2909.11
35	Platelets (G/L)	1.71–459000	scale	3	113206.44	107118.63
36	Albumin (mg/dL)	1.9–4.9	scale	6	3.45	0.69
37	Total Bilirubin(mg/dL)	0.3–40.5	scale	5	3.09	5.50
38	Alanine transaminase (U/L)	11–420	scale	4	67.09	57.54
39	Aspartate transaminase (U/L)	17–553	scale	3	96.38	87.48
40	Gamma glutamyl transferase (U/L)	23–1575	scale	3	268.03	258.75
41	Alkaline phosphatase (U/L)	1.28–980	scale	3	212.21	167.94
42	Total Proteins (g/dL)	3.9–102	scale	11	8.96	11.73
43	Creatinine (mg/dL)	0.2–7.6	scale	7	1.13	0.96
44	Number of Nodules	0–5	scale	2	2.74	1.80
45	Major dimension of nodule (cm)	1.5–22	scale	20	6.85	5.10
46	Direct Bilirubin (mg/dL)	0.1–29.3	scale	44	1.93	4.21
47	Iron (mcg/dL)	0–224	scale	79	85.60	55.70
48	Oxygen Saturation (%)	0–126	scale	80	37.03	28.99
49	Ferritin (ng/mL)	0–2230	scale	80	439.00	457.11
50	Class	0, 1	binary	0		

Normalised Ratio(0.079), Gamma glutamyl transferase (U/L)(0.075), Alpha-Fetoprotein (ng/mL)(0.056), Oxygen Saturation (%)(0.056), Platelets (G/L)(0.044). The significance values for all features are available in the supplementary materials.

Table 1 provides details about the characteristics in the data set and information about missing values.

1.2. Goals

The main goals of this study are as follows:

- introduction of a new logistic regression model training method, instead of iteratively reweighted least squares, genetic algorithms are used;
- fusion of genetic algorithms for both training the logistic regression model and feature selection;
- verification of solutions implemented and comparison against standard learning methods.

1.3. Related works

Many machine learning models have been prepared to detect of liver cancer, especially hepatocellular carcinoma, in recent years. Santos et al. [41] proposed a new cluster-based oversampling approach for HCC detection based on the K-means clustering and SMOTE algorithm to build a representative data set. In this study, logistic regression and neural networks were also used. The best model achieved a classification efficiency: 75.19%. Sawhney et al. [44] proposed a method based on the firefly algorithm and random forest to detect several types of cancer. In the case of hepatocellular carcinoma, the proposed model had an accuracy of 83.5%. Ksiazek et al. [26] designed a machine learning model based on the support vector machine and genetic algorithms. Genetic algorithms were used to optimize both the classifier's parameters and feature selection. The proposed model obtained a high classification accuracy of 88.49%. Nayak et al. [31] prepared a classification model enabling the detection of hepatocellular carcinoma based on CT images. He proposed to use SVM with the RBF kernel. The best result was 86.9%. Brehar et al. [13] designed a classification model for HCC detection based on ultrasound images. The model was prepared with the use of AdaBoost and achieved a classification accuracy of 72%. A combination of the support vector machine together with the Lasso method was proposed by Aonpong et al. [9]. The research was conducted on a data set of 331 patients from Sir Run Run Shaw Hospital, Zhejiang University, China. The proposed model achieved a classification accuracy of 89.18%. Research using decision trees as well as linear regression and boosting was conducted by Hashem et al. [19]. Their work was based on a data set of over 4000 patients obtained from the Egyptian National Committee for the Control of Viral Hepatitis and the multidisciplinary HCC clinic at Cairo University's Kasr Al-Aini Hospital. The best proposed model was characterised by a very high classification accuracy: 95.6%. A new method for HCC detection was proposed by Tuncer et al. [47]. It is based on a neighbourhood component analysis and a relief based method. Using FGSVM, a very high accuracy was obtained: 92.12%. Sato et al. [43] conducted hepatocellular cancer detection studies in two groups (539 and 1043 patients). They used the following classifiers: logistic regression, support vector machine, gradient boosting, random forest, neural network and deep learning. The best result was obtained with the use of the boosting gradient - 87.34%. In order to detect this disease, research was also carried out using gene expression profiles data sets. Zhang et al. [48] conducted their experiments with the support vector machine and achieved very high results. Ali et al. [7] proposed the LDA-GA-SVM method combining the LDA method for dimensional reduction, genetic algorithms and a support vector machine. The proposed classifier achieved an accuracy of 90.30%. A model with exactly the same accuracy was proposed by Książek et al. [27]. A model of ensemble learning based on stacking learning was designed, consisting of 7 classifiers and a meta-classifier. Genetic algorithms were also used to optimize parameters and select features of individual classifiers. Hattab et al. [20] proposed a new approach using the k-means algorithm, SMOOTE method and SVM. Their model achieved a classification efficiency of 84.90%. Al-Islam et al. also used the SMOOTE technique. Combining it with the XGBoost algorithm, they achieved a high detection efficiency of hepatocellular carcinoma equal to 87%. For the detection of liver disease, Abdar et al. [5] proposed a regression tree (Cart) with a boosting technique, as well as a multi-layer perceptron neural network (MLPNN). It also implemented a combination of the two MLPNNB-C5.0 methods as well as the MLPNNB-CHAID algorithm. Deep learning methods described in detail in the works were also used to detect liver cancer: [12,14,15].

2. Methods

This section presents all the steps taken to achieve the introduced results. Firstly, we discuss the scaling methods, later on filling in missing values. In the end, genetic algorithms are introduced, which are being used to calculate logistic regression coefficients as well as to select the best set of features.

2.1. Preprocessing

Since machine learning algorithms demand the data to be complete, there are a few preprocessing scenarios tested. A vast amount of missing values present in the processed dataset (Table 1), require additional investigation to prepare the data for further analysis. Furthermore, the data have a large numerical spread. To scale data, the standardisation method was applied. It must be noted that in machine learning preprocessing plays a crucial role in optimizing the classifier's performance and accuracy. Without using proper preprocessing the output model can be easily over-fitted and have a poor generalisation ability. Preprocessing also impacts the classifier's training and prediction performance. It is crucial when fusing it with genetic algorithms since lot of different classifiers with different parameters and features must be trained in order to achieve the best results.

2.1.1. Scaling

- **standardisation** was calculated for each feature separately, using a standard formula:

$$scaled\_value = \frac{x - \overline{u}}{s} \tag{1}$$

where:

- x - a sample
- $\overline{u}$ - mean of training samples for a single feature
- s - a standard deviation of training samples for a single feature

Preprocessing is applied to both ordinal (including binary) and numerical values.

2.1.2. Missing values

The amount of missing values in each feature is presented in Table 1. Corresponding to the referenced table the count of missing samples varies between 0 and 80 for a single feature. Moreover, a different type of data is to be noticed (binary, ordinal, scale) which forces usage of different preprocessing scenarios. Two methods of filling the missing values were tested:

- Samples of the *binary* and *ordinal* type were replaced using mode, samples of *scale* type were replaced using mean for each feature separately.

- Additionally, a more advanced method of filling in the missing values was tested: the k-nearest neighbours algorithm. The experiment was carried out for 3 and 5 neighbours using the Euclidean metric [32].

2.2. Cross-validation

For all the presented experiments, the stratified five-fold CV is used. Each fold has randomly selected train and test sets with the original proportions between classes preserved. Preserving the number of instances from all the classes in each set is crucial in unbalanced data sets processing. Cross-validation helps the model to over-fit.

2.3. Genetic algorithms

Genetic algorithms are algorithms based on the fundamental laws of evolution [21,29,40]. Each individual in a population is a potential solution to the problem. The algorithm is iterative - during the subsequent epochs, individuals are selected, crossed and mutated. After each iteration, new individuals are created as a result of crossover and mutation. The next generation is created, which is later on assessed again using the same adaptation function. The algorithm ends when the specified accuracy is reached or after a given number of epochs. Genetic algorithms have also found application in machine learning problems. They were used to optimize parameters and select features in hepatocellular carcinoma detection [26,27], ECG signal [35,38], credit scoring [37] or the detection of heart diseases [2,8,10]. The single individual in the population is, in this case, a single set of parameters for the machine learning model. Additionally, this individual can be expanded with the set of features from the data set. It is assessed through classic machine learning metrics such as accuracy, or the f1-score presented in section 2.4. Other metrics as specificity, sensitivity or AUC can also be applied. We decided to use the f1-score and accuracy because they are the most frequently used in the literature and therefore, we can compare our results to the results of other authors.

The proper configuration of the genetic algorithm is significant. Depending on the problem to be solved, it can be very different. Table 2 shows the setup used in our experiments. The particular values have been selected based on trial and error but also based on our experience. The high mutation probability ensures high variance in the whole population. An elitist strategy ensures that the best individual is automatically moved to the next epoch without crossover and mutation. It is crucial in keeping the constant convergence of the genetic algorithm. The best potential solution can be easily lost due to the high probability of mutation and crossover. In order to compare the results achieved by the genetic algorithm, a reference experiment was carried out using the PSO algorithm [25]. We set the number of iterations and individuals to 2000 for PSO. Additionally, we have configured omega = 0.2, phip = 0.2, phig = 0.2.

Table 2
Genetic algorithm parameters [27].

Parameter	Value
Crossover algorithm	Two points crossover or arithmetic crossover
Selection algorithm	Tournament selection (with three individuals participating in each tournament)
Mutation algorithm	Own implementation of a single point mutation in each experiment
Probability of crossover	0.7
Probability of mutation	0.7
Population size	2000
Number of iteration	2000
Elitist strategy	The best individual goes to the next iteration
Fitness function	Accuracy or F1-score

2.4. Metrics

For the purpose of evaluating the model's performance, standard metrics were chosen: Accuracy (ACC) and F-measure (f1-score), Sensitivity, Specificity, Brier Score. All metrics were calculated based on the confusion matrix generated for each experiment separately [27]. Those metrics are used to show the final results of the classification.

3. Logistic regression

Logistic Regression is a mathematical model which enables the probability estimation of belonging to a certain class. In this paper, the LR model is used for binary classification, but in other cases, it can easily be extended for multi-label classification.

To calculate regression coefficients, usually, iterative methods are used such as iteratively reweighted least squares (IRLS) or the Newton-Raphson method. In this study, those common methods are replaced with genetic algorithms.

The logistic regression model implementation used in this study comes from the Sklearn library [33]. It allows for easy control of a few parameters, such as:

- Penalty - adds bias to the model when it is suffering from high variance,
- C - the higher values generalize the model, whereas the smaller values constrain it more,
- Max iteration - the maximum number of iterations done in order to converge the model,
- dual - an objective function type,
- fit_intercept - increase or decrease the impact of an intercepted value,
- solver - a type of an algorithm solver,
- l1_ratio - controls the penalty impact.

All those parameters are tuned using genetic algorithms. The details are presented in section 4.

4. Experiments

As part of this research, experiments were conducted on a data set collected at Coimbra's Hospital and University Center (CHUC), Portugal. The data set is described in detail in section: 1.1. In the first stage of the experiment, the missing values were completed. For quantitative attributes, an average is used, and for qualitative attributes, a mode value. The entire experiment was performed using 5-fold cross-validation. The accuracy and f1-score were selected as the basic metrics. The research was divided into 3 stages:

- Genetic optimization of parameters
- Genetic optimization of parameters and selection of features
- Genetic weight optimization

These experiments are described in detail in the sections: 4.1, 4.2 and 4.3. Fig. 1 shows the experiment schematics. The three experiments mentioned will be described in detail later in this chapter.

4.1. Genetic parameters optimization

In the first experiment, genetic algorithms were used to optimize the parameters of the logistic regression model.

Table 3 shows the parameters of the logistic regression model available to tune. These parameters include C, Max iteration, Penalty, Dual, Fit intercept, solver, and L1 ratio. Each individual in the genetic population consists of a chromosome containing the parameters mentioned above. The values are randomly generated within the specified ranges to create the initial population.

Fig. 2 shows the structure of the chromosome for an exemplary

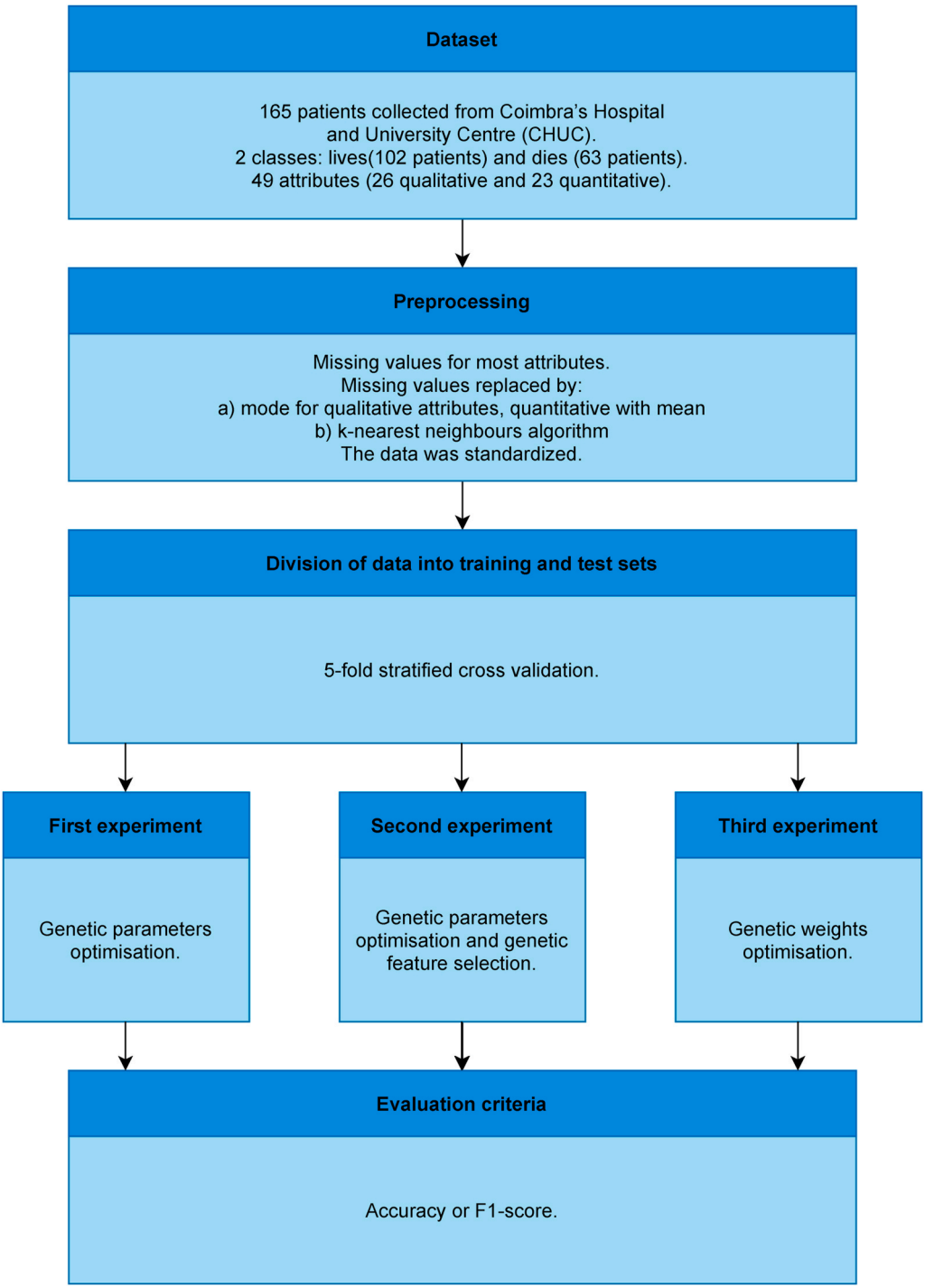

Fig. 1. Experiment schema.

Table 3
Logistic regression parameters.

Parameter	Value
C	[1–100]
Max iteration	[1–2000]
Penalty	[L1, L2, Elasticnet, none]
Dual	True or False
Fit intercept	True or False
Solver	[newton-cg, lbfgs, liblinear, sag, saga]
L1 ratio	[0–1]

individual. It consists of 7 genes.

4.2. Genetic parameters optimization and feature selection

The second experiment extends the first one with the use of genetic algorithms to select the features. It is widely known that many machine learning algorithms perform better on fewer features. However, it is not a simple task to choose the optimal set of features. In this experiment, this selection will be made by evolutionary algorithms.

Table 4 presents in detail the parameters of the implemented model. As in the first experiment, these are logistic regression parameters, as well as 49 parameters, each corresponding to one feature available in

C	Max iteration	Penalty	Dual	Fit intercept	Solver	L1 ratio
23.5	1500	L1	True	False	Lbfgs	0.55

Fig. 2. Example chromosome in experiment 1.

Table 4
Model parameters in experiment 2.

Parameter	Value
C	[1–100]
Max iteration	[1–2000]
Penalty	[L1, L2, Elasticnet, none]
Dual	True or False
Fit intercept	True or False
Solver	[newton-cg, lbfgs, liblinear, sag, saga]
L1 ratio	[0–1]
Feature 1	0-feature rejected, 1 – feature accepted
Feature 2	[0–1]
Feature …	[0–1]
Feature 49	[0–1]

the dataset. If the value of the parameter equals 1, the analogical feature is used to train the classifier when it is 0.

Fig. 3 shows the structure of the chromosome used in experiment number 2. The structure of a chromosome consists of logistic regression parameters and features from the data set. In total, the chromosome consists of 56 genes. Therefore, the optimization problem becomes much more complex compared to the one introduced in section 4.1.

4.3. Genetic weights optimization

The last experiment presents a novel approach to calculate the logistic regression coefficients. The weights optimization was performed using genetic algorithms, contrary to the most common method like IRLS (gradient algorithms). This approach provided the highest accuracy.

Table 5 shows the detailed parameters of the model in the experiment on optimizing logistic regression weights. In this case, this algorithm's classic parameters, such as max iteration, C or solver, are not optimized - because the goal of the experiment is to use genetic algorithms to select an appropriate set of weights. In the standard approach, it is done by algorithms that are implemented in the Sklearn library. The amount of weights is equal to the number of features in dataset 49.

Fig. 4 shows the structure of the chromosome used in experiment 3. This chromosome consists only of logistic regression weights. There are 49 values present in the chromosome, each in the range [-3, 3]. This constructs a much more complex optimization problem which has to be solved.

5. Results

In this section, the results obtained with the models proposed in sections 4.1, 4.2 and 4.3 will be presented.

The models were implemented using Python (version 3.8) with libraries: Sklearn [33], Pandas, PySwarm and Deap [17].

The calculations were performed on a machine with the following specifications:

Table 5
Model parameters in experiment 3.

Parameter	Value
Weight 1	[-3,3]
Weight 2	[-3,3]
Weight …	[-3,3]
Weight 49	[-3,3]
Intercept	[-3,3]

Weight 1	Weight 2	Weight ...	Weight 49	Intercept
1.24	-2.8	...	-0.7	0.22

Fig. 4. Example chromosome in experiment 3.

- Processor: Intel(R) Xeon(R) CPU E5-2670 v3 @ 2.30 Ghz 2.30 Ghz (two processors)
- RAM: 512 GB
- Operating system: Windows Server 2019 (64-bits)

Due to the high amount of epochs used to optimize the model, the computation times on the specified machine took around 2 h to complete for a single experiment. The calculations were performed using all available cores. That made it possible to speed up the calculations compared to common one-core solutions.

For each-cross validation fold the presented results are calculated on different test sets for each fold, which is not used during the classifiers training. At each cross-validation iteration, 132 samples formed the training set, and the remaining 33 built the test set. 12 samples from class 0 and 21 samples from class 1 formed a test set, where 51 samples from class 0 and 81 from class 1 formed a training set.

5.1. Genetic parameter optimization

In this experiment, the model detailed in section 4.1 was extended to optimize logistic regression parameters genetically. The results for two different target functions of the genetic algorithm: accuracy and f1-score, are presented below.

5.1.1. Model with accuracy optimization

The model prepared in this experiment achieved a classification accuracy of 78.79%. It was a model that used the liblinear gradient algorithm to solve the problem of selecting logistic regression weights. The best result was achieved after 25 iterations. The model using the training set achieved an accuracy equal to: 90.15%.

The detailed parameters of the model are presented in Table 6. The experiment was performed using 5-fold cross-validation.

C	Max iteration	Penalty	Dual	Fit intercept	Solver	L1 ratio	Feature 1	Feature 2	Feature ...	Feature 49
23.5	1500	L1	True	False	Lbfgs	0.55	0	1	...	1

Fig. 3. Example chromosome in experiment 2.

Table 6
Model parameters in experiment 1 with accuracy as a fitness function.

Parameter	Value
C	4.0753
Max iteration	25
Penalty	l2
Dual	True
Fit intercept	True
Solver	liblinear

Table 7 shows the detailed classification results for the proposed model. In this experiment, the k-nearest neighbours algorithm (3 neighbours) was chosen to fill in the missing values. The average classification accuracy in individual folds is 78.79%.

Fig. 5 shows the ROC curve for the proposed model. For both class 1 and class 2, the area under the curve is 0.79.

5.1.2. Model with f1-score optimization

In this part of the experiment, the model proposed in section 4.1 was re-used. This time the matching function in the genetic algorithm has been changed. The f1-score was used instead of the accuracy measurement. In this part of the experiment, the missing values were completed using the k-nearest neighbours algorithm with 5 neighbours. The obtained results are similar to those in section 5.1.1. The best result obtained by the model was the f1-score equal to 78.71%. In order to verify the model overfitting, it was tested on the training set. The achieved f1-score was 89.88%.

This time the best model was obtained with the use of the solver 'liblinear'. The algorithm took 9 iterations to achieve convergence. Detailed parameters of the presented model are available in Table 8.

Table 9 shows the detailed results of the model under the individual cross-validation folders.

Fig. 6 shows the ROC curve for the model used in this experiment. The AUC area under the ROC curve is 0.81.

5.2. Genetic parameters optimization and feature selection

This experiment is a continuation of the experiment described in section 4.1. Here it is additionally extended with a genetic selection of features. A detailed description of this experiment is presented in section 4.2. As in the previous experiment, the model was optimized once with the use of accuracy as a fitness function - section 5.2.1, and then the f1-score - section 5.2.2.

5.2.1. Model with accuracy optimization

In this section, the results will be presented for the accuracy as the fitness function. This time the optimization problem is more complex - apart from the selection of the logistic regression parameters, it was also necessary to choose the best set of features. In this experiment, the knn algorithm was again used to fill in the missing values with the number of neighbours set to 3. This task was solved - the proposed model achieved a classification accuracy of 89.7%. However, the accuracy achieved on the training set is 89.09%.

Table 10 shows the parameters and a set of features for a given

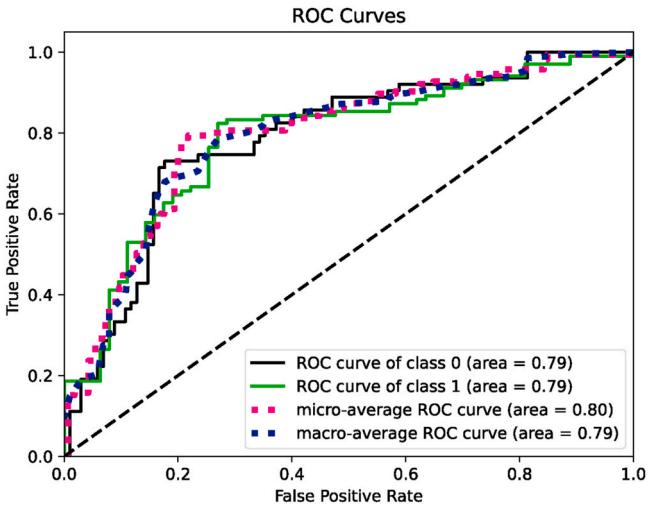

Fig. 5. The ROC curve in experiment 1 for the model with accuracy.

Table 8
Model parameters in experiment 1 with an f1-score as a fitness function.

Parameter	Value
C	1.8815
Max iteration	9
Penalty	'l1'
Dual	False
Fit intercept	False
Solver	'liblinear'
L1 ratio	0.8959

model. This time, the 'liblinear' algorithm was used as a solver, which achieved convergence already in epoch 111. Additionally, the set of features has been reduced from 49 to 25.

Table 11 shows the detailed results of the experiment performed. The classification accuracy of over 90% can be seen in as many as 3 folds out of 5. Additionally, the set of features necessary to build an effective model has been significantly reduced.

Fig. 7 shows the ROC curve for the proposed model. Much higher AUC value is to be noticed compared to the models from section 4.1. This time it is 0.88.

5.2.2. Model with f1-score optimization

In this model, the F1-score measure was used as the fitness function. The knn algorithm for missing values has been configured with a number of neighbours of 5. The result obtained was very similar to the result from section 5.2.1. The model obtained an f1-score: of 88.31%. The f1-score value on the training set was: 86.33%.

Table 12 shows the best set of parameters for this model. The algorithm converged in 181 iterations with the 'lbfgs' algorithm. As in the case of the model in section 5.1.1, 23 features were used to achieve the best result.

Table 13 shows the detailed results of the experiment performed.

Table 7
Detailed results for the model with an optimized accuracy in experiment 1.

Classifier	Fold	TP	TN	FP	FN	Sen	Spec	Brier	Acc
Logistic regression	1	6	20	6	1	0.5	0.9524	0.2092	0.7879
	2	9	19	3	2	0.75	0.9048	0.1309	0.8789
	3	11	14	2	6	0.8462	0.7	0.2396	0.7576
	4	12	15	1	5	0.9231	0.75	0.2116	0.8182
	5	7	17	6	3	0.5385	0.85	0.2367	0.7273
	Total					0.7115	0.8314	0.2056	**0.7879**

Table 9
Detailed results for the model with an optimized f1-score in experiment 1.

Classifier	Fold	TP	TN	FP	FN	Sen	Spec	Brier	F1-score
Logistic regression	1	9	19	3	2	0.75	0.9048	0.154	0.8332
	2	10	19	2	2	0.8333	0.9048	0.1067	0.869
	3	11	13	2	7	0.8462	0.65	0.2227	0.7263
	4	13	11	0	9	1.0	0.55	0.2489	0.7263
	5	10	16	3	4	0.7692	0.8	0.195	0.7806
	Total					0.8397	0.7619	0.1855	0.7871

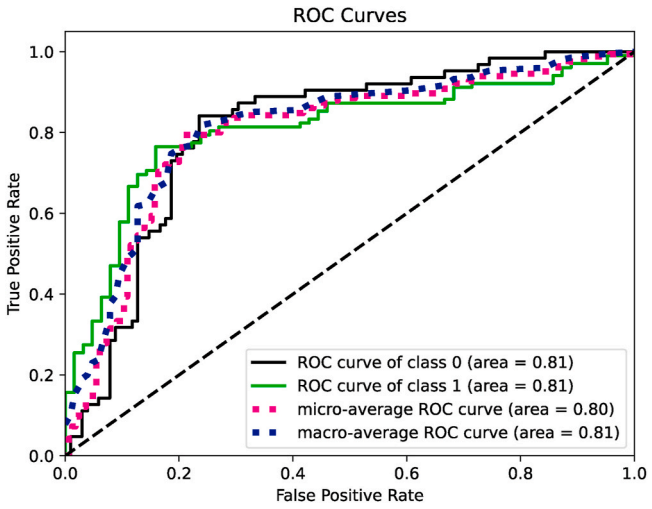

Fig. 6. The ROC curve in experiment 1 for the model with f1-score.

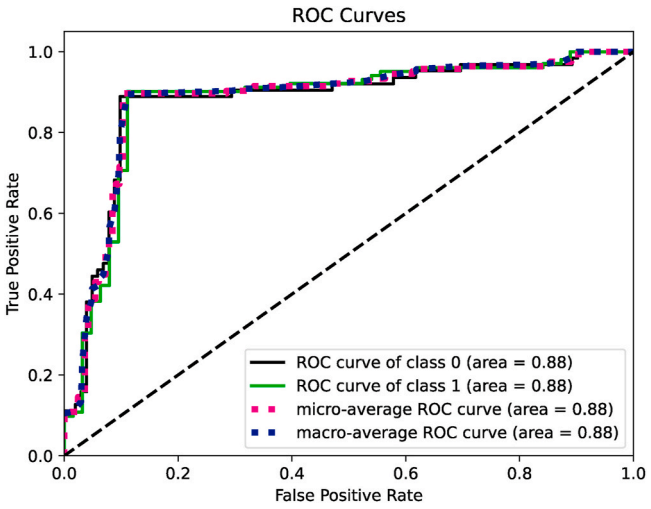

Fig. 7. The ROC curve in experiment 2 for the model with accuracy.

Table 10
Model parameters in experiment 2 with accuracy as a fitness function.

Parameter	Value
C	13.6838
Max iteration	111
Penalty	'l2'
Dual	False
Fit intercept	False
Solver	'liblinear'
Selected features:	Gender, Symptoms, Hepatitis B e Antigen: HBeAg, Hepatitis B Core Antibody: HBcAb, Hepatitis C Virus Antibody: HCVAb, Endemic Countries, Diabetes, Arterial Hypertension: AHT, Human Immunodeficiency Virus: HIV, Nonalcoholic Steatohepatitis: NASH, Age at diagnosis, Grams of Alcohol per day: Grams/day, Encefalopathy degree, International Normalised Ratio: INR, Leukocytes(G/L), Platelets (G/L), Total Bilirubin(mg/dL): Total Bil, Alanine transaminase (U/L): ALT, Aspartate transaminase (U/L): AST, Alkaline phosphatase (U/L): ALP, Total Proteins (g/dL): TP, Creatinine (mg/dL), Major dimension of nodule (cm), Iron (mcg/dL), Ferritin (ng/mL)

Again, high F1-scores in individual folds are to be noticed, which translates into a high final, average result.

Fig. 8 shows the ROC curve for the proposed model. The area under the curve is identical to the model with accuracy and amounts to 0.86.

Table 12
Model parameters in experiment 2 with an f1-score as a fitness function.

Parameter	Value
C	91.7618
Max iteration	181
Penalty	'None'
Dual	False
Fit intercept	True
Solver	'lbfgs'
Selected features:	Gender, Hepatitis B Surface Antigen: HBsAg, Hepatitis B e Antigen: HBeAg, Hepatitis C Virus Antibody: HCVAb, Diabetes, Arterial Hypertension: AHT, Chronic Renal Insufficiency: CRI, Portal Hypertension, Liver Metastasis, Radiological Hallmark, Age at diagnosis, Ascites degree, International Normalised Ratio: INR, Platelets (G/L), Total Bilirubin(mg/dL): Total Bil, Alanine transaminase (U/L): ALT, Aspartate transaminase (U/L): AST, Gamma glutamyl transferase (U/L): GGT, Total Proteins (g/dL): TP, Creatinine (mg/dL), Major dimension of nodule (cm), Iron (mcg/dL), Ferritin (ng/mL)

5.3. Genetic weights optimization

The last part of the experiment shows the results for the model proposed in section 4.3. This is a novel approach to find weights of the logistic regression model. Evolutionary computations were used instead of usual weight optimization with gradient methods. This allowed for

Table 11
Detailed results for the model with optimized accuracy in experiment 2.

Classifier	Fold	TP	TN	FP	FN	Sen	Spec	Brier	Acc
Logistic regression with genetic parameteroptimization and genetic feature selection	2	12	21	0	0	1	1	0.0448	1
	3	12	19	1	1	0.95	0.9231	0.1205	0.9394
	4	11	16	2	4	0.8	0.8462	0.1745	0.8182
	5	10	17	3	3	0.85	0.7692	0.1785	0.8182
	Total					0.901	0.891	0.1287	0.897

Table 13
Detailed results for the model with an optimized f1-score in experiment 2.

Classifier	Fold	TP	TN	FP	FN	Sen	Spec	Brier	F1-score
Logistic regression with genetic parameter optimization and genetic feature selection	1	11	20	1	1	0.9524	0.9167	0.1238	0.9345
	2	11	21	1	0	1.0	0.9167	0.0619	0.9666
	3	10	17	3	3	0.85	0.7692	0.1813	0.8096
	4	11	18	2	2	0.9	0.8462	0.1383	0.8731
	5	9	17	4	3	0.85	0.6923	0.1822	0.7746
	Total					0.9105	0.8282	0.1375	0.8717

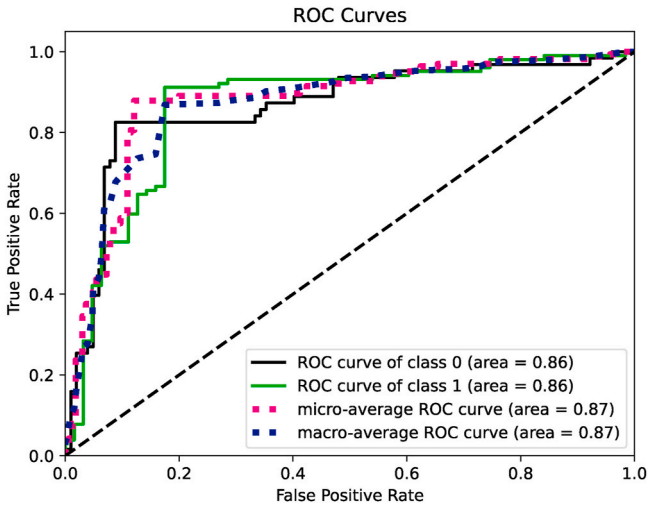

Fig. 8. The ROC curve in experiment 2 for the model with an f1-score.

obtaining new, high results. As in the previous experiments, this section will present the results for optimizing the two fitness functions.

5.3.1. Model with accuracy optimization

This subsection describes the results for the model optimized for accuracy. In this experiment, we optimize the logistic regression weights using all the features from the data set. The missing values were filled in using the k-nearest neighbours algorithm, where $k = 3$. The best result is higher than the results from previous experiments. The accuracy achieved is 94.55%. The same accuracy was achieved on the training set.

Table 14 shows the detailed calculation results for the model with weight optimization. Very high scores can be observed in each cross-validation fold, above 90%.

Table 15 shows the weight and intercept values used in the logistic regression model within the experiment.

Fig. 9 shows the ROC curve for the implemented model. The AUC value is very high: 0.92. The AUC is much greater than in previous experiments.

5.3.2. Model with f1-score optimization

An analogous experiment of weight optimization with the use of genetic algorithms was performed for the f1-score. Again, very high results were achieved. The f1-score was 93.56%. A similar F1-score value was achieved on the training set: 93.65%.

Table 16 shows the exact results of the experiment performed. High f1-scores can be observed in each cross-validation fold.

Table 17 shows the weight and intercept values used in the logistic regression model within the experiment.

Fig. 10 shows the ROC curve for the optimized model. Again, a much higher AUC value can be observed than in the previous experiments for the f1-score. This value is 0.93.

5.4. The impact of filling the missing values technique

One of the most important obstacles to be solved before starting the design of a machine learning model is filling the missing values in the dataset. The easiest techniques require filling them using mode or mean values. Results for the experiment with the use of mode and mean are available in supplementary materials. However, the use of more advanced techniques such as the k-nearest neighbours algorithm can significantly improve the results. In our case, thanks to the mentioned algorithm, it was possible to improve the final results in 5 out of 6 conducted experiments.

Table 18 shows that the use of the nearest neighbours algorithm significantly improved the results achieved. Especially significant improvement is to be noticed in the first experiment - genetic selection of parameters.

5.5. Comparison of genetic algorithms with PSO

The experiment compared two very popular biology-inspired methods for solving the optimization problem: evolutionary algorithms and PSO.

By analyzing Table 19, one can observe that the genetic algorithms achieved better results. It is especially visible in the case of the last experiment - weight optimization. That problem was particularly difficult to optimize, because 66 parameters had to be selected (having values on a continuous scale) in the range [-3.3]. In the case of experiments 1 and 2, which were simpler in terms of optimization, the results achieved by PSO were less divergent compared to the results obtained with the use of evolutionary calculations. A detailed analysis of the results obtained from genetic algorithms and PSO is provided in the supplementary materials.

6. Discussion

In this article, we focused on different approaches to fuse genetic algorithms with a logistic regression model. A proper preprocessing demanded by the investigated dataset is designed. It includes filling in

Table 14
Detailed results for the model with optimized accuracy in experiment 3.

Classifier	Fold	TP	TN	FP	FN	Sen	Spec	Brier	Acc
Logistic regression with genetic weights optimization	1	10	20	2	1	0.9524	0.8333	0.1005	0.9091
	2	12	21	0	0	1.0	1.0	0.0303	1.0
	3	12	18	1	2	0.9	0.9231	0.097	0.9091
	4	13	18	0	2	0.9	1.0	0.0862	0.9394
	5	12	20	1	0	1.0	0.9231	0.0541	0.9697
	Total					0.9505	0.9359	0.0736	**0.95**

Table 15
Logistic regression coefficients for each feature with intercept for an ACC measure.

weights	−0.4843	−1.7421	−0.4282	0.9403	0.0129	−0.6173	−1.9304
	0.3465	1.1663	1.8194	−1.3592	0.2901	−0.5994	0.8925
	0.5636	−1.4497	−1.1905	0.4035	−1.6433	0.9284	−1.774
	−0.7573	0.7663	−2.7800	−0.7964	−1.5463	−1.6388	−1.802
	0.3005	−1.2226	1.7008	−0.6957	0.4234	−1.0089	1.7387
	0.7882	−1.335	1.3087	−1.7784	−0.6852	−1.9369	−0.1891
	−1.5178	1.0284	−0.6915	−1.7842	1.9373	1.7000	−1.2787
intercept	0.8192						

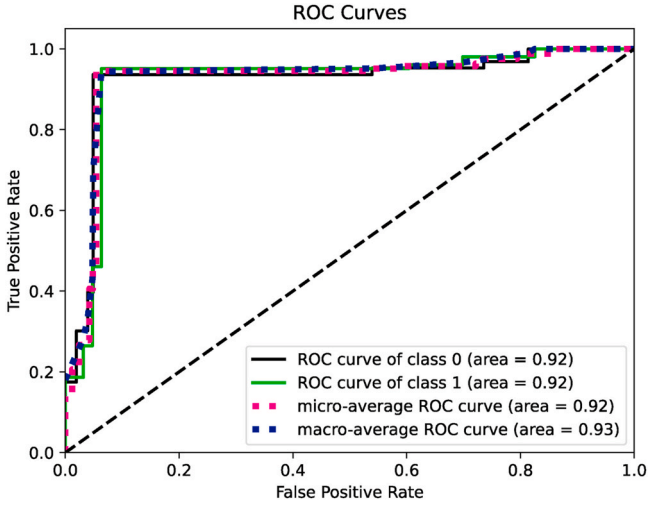

Fig. 9. The ROC curve in experiment 3 for the model with accuracy.

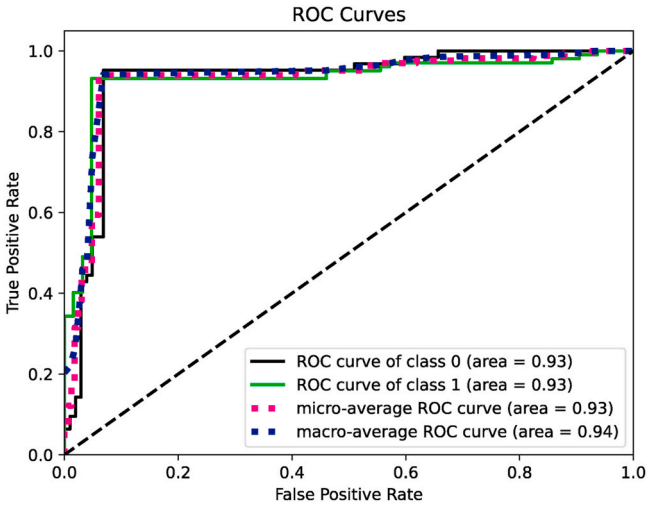

Fig. 10. The ROC curve in experiment 3 for the model with an f1-score.

missing values and data scaling. We conducted three different experiments showing how those algorithms can be combined. A general overview of the experiment is presented in Fig. 1. The experiments were carried out on the dataset collected in Coimbra's Hospital and University Center (CHUC), Portugal. A detailed description of the data set is provided in section 1.1.

In the first experiment, we optimized the logistic regression parameters using evolutionary calculations. This experiment is detailed in section 4.1. The results for this experiment are described in detail in section 5.1, which is divided into two sections, 5.1.1 and 5.1.2 - depending on the optimized objective function (accuracy or f1-score). The model from the first experiment achieved a classification accuracy

Table 18
Comparison of results for different approaches to fill in missing values.

Experiment	Missing value algorithm	Accuracy [%]	F1-score [%]
Genetic parameter optimization	Mean and mode	76.36	75.14
	KNN	78.79	78.71
Genetic parameters and feature selection	Mean and mode	89.09	88.31
	KNN	89.7	87.17
Genetic weights optimization	Mean and mode	93.94	91.83
	KNN	94.55	93.56

Table 16
Detailed results for the model with an optimized f1-score in experiment 3.

Classifier	Fold	TP	TN	FP	FN	Sen	Spec	Brier	F1-score
Logistic regression with genetic weights optimization	1	9	19	3	2	0.9048	0.75	0.1652	0.8332
	2	12	21	0	0	1.0	1.0	0.0173	1.0
	3	13	13	0	2	0.9	1.0	0.0951	0.938
	4	13	19	0	1	0.95	1.0	0.062	0.9687
	5	13	18	0	2	0.9	1.0	0.105	0.938
	Total					0.931	0.95	0.0889	**0.95**

Table 17
Logistic regression coefficients for each feature with intercept for the f1-score measurement.

weights	0.7485	−1.9286	1.0083	1.3137	1.1659	−1.5984	−1.909
	−1.3562	1.3241	0.5996	−2.3965	1.0252	−0.1267	2.8711
	−1.9769	−1.5686	0.5767	0.1817	−0.7776	1.0228	−0.6392
	0.4413	1.4618	−1.8221	−1.3657	−1.1935	−1.7647	−0.728
	0.2569	−1.4082	0.3251	1.0778	1.0223	−0.7939	0.9067
	1.9502	−0.4915	1.791	−1.9927	−0.7666	−0.8295	0.1534
	1.2789	−0.0325	−1.1064	−1.9777	1.4487	−0.0712	−1.8004
intercept	1.7241						

Table 19
Comparison of the results achieved from using GA and PSO.

Experiment	Optimization	Accuracy [%]	F1-score [%]
Genetic parameter optimization	GA	78.79	78.71
	PSO	77.57	78.07
Genetic parameters optimization and feature selection	GA	89.7	87.17
	PSO	86.66	85.98
Genetic weights optimization	GA	94.55	93.56
	PSO	89.09	87.05

of 78.79% (detailed results are presented in Table 7) and an f1-score of 78.71% (detailed results are available in Table 9). Tables 6 and 8 present the parameters of the best models after the optimization process. Figs. 5 and 6 show the ROC curves obtained in this experiment. The areas under these curves are 0.77 and 0.79 and 0.81, respectively. Experiment 2 is an extension of experiment 1 with a genetic selection of features. The selection of appropriate attributes significantly improves the classification results in most machine learning models. This experiment is detailed in section 4.2. Section 5.2 presents the results of this experiment. As in the case of the first experiment, the model was optimized twice - firstly, when the target function was accuracy (section 5.2.1), secondly, for the F1-score function (section 5.2.2). Genetic selection made it possible to significantly improve the results for both the model with optimized accuracy (Table 11) and the f1-score (Table 13). The best accuracy achieved was 89.7%, while the f1-score's result was 87.17%. The parameters for these models are presented in Tables 10 and 12. ROC curves were also prepared for this experiment - they are presented in Figs. 7 and 8. The AUC area under those curves was 0.88 and 0.86. The last experiment is a new approach to fit logistic regression model. In this approach, the classic gradient learning method was replaced with genetic algorithms - the logistic regression weights were selected through evolutionary computations. This experiment is detailed in section 4.3. As done for previous experiments, the results are presented in section 5.3 (divided into two fitness functions: accuracy - section 5.3.1 and f1-score - section 5.3.2). Thanks to this innovative approach, very high results were achieved. The accuracy of the classification was: 94.55%, and the f1-score was 93.56%. Detailed results for this experiment are shown in Tables 14 and 16. The ROC curves (Figs. 9 and 10) were again prepared - with values of AUC 0.92 and 0.93, respectively.

A summary of the obtained results is shown in Fig. 11. We observed that with subsequent experiments, the obtained results were rising. The best result was achieved by using the novel approach with a genetic optimization of weights. This shows another application of evolutionary computing in optimization problems - applied to weights optimization.

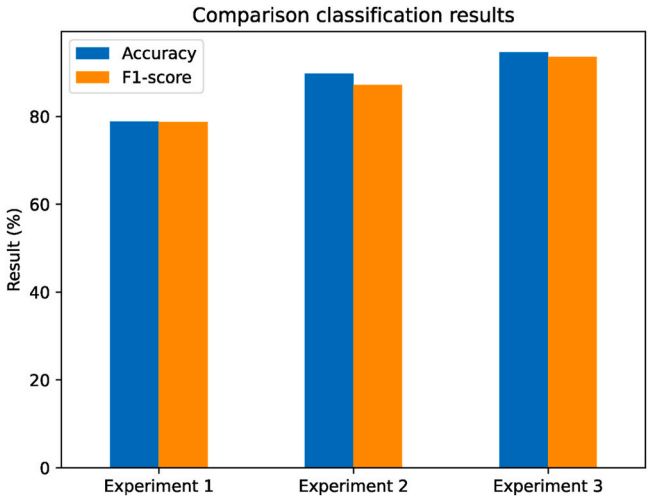

Fig. 11. Summary of the obtained results.

Table 20
Comparison of the results obtained in the HCC detection problem on the CHUC dataset.

Study	Method	Accuracy	F1-score
Santos et al.	NN + augmented set approach	0.7519 +- 0.0105	0.6650 +- 0.0182
Sawhney et al.	BFA + RF	0.835	N. A
Ksiazek et al.	SVC with new 2-level genetic optimizer approach	0.8849	0.8762
Ali et al.	LDA–GA–SVM (with linear and RBF kernel)	0.9030	N. A
Książek et al.	StackingGA	0.9030	0.8857
Hattab et al.	K-Means + SMOOTE + SVM	0.8490	N. A
Al-Islam et al.	SMOOTE + XgBoost	0.87	N. A
Tuncer et al.	reliefF + LDA	0.8303	0.8202
	NCA + FGSVM	0.9212	0.9161
This study	**GA-LR**	0.9455	0.9356

In the literature, a few papers described the detection of hepatocellular carcinoma in the CHUC dataset. The collected results are presented in Table 20. Various methods to solve the problem of hepatocellular carcinoma detection can be noticed in the scientific papers. Paper [41] based on the use of neural networks obtained results at the level of 70%. The use of genetic algorithms to optimize models [7,26,27] allowed for a significant improvement in the results achieved - the best accuracy of such models is 90.30%. In Ref. [47] the authors proposed a completely new approach built on the NCA and the relief-based method. The classification accuracy obtained in this paper equals to 92.12%. The papers [20,22] use the SMOOTE algorithm with various classifiers. This made it possible to obtain models with an accuracy of over 80%. Our research is significantly different from the rest. Even though a very popular logistic regression algorithm was used, it has been linked to genetic algorithms in three different ways. Testing various fusion scenarios of these two algorithms allowed us to obtain very high results, achieving the best result in all of the literature - a classification accuracy equal to 94.55%, and at the same time a high value of f1-score 93.56%. The main advantages of the proposed model are:

- three approaches to fuse logistic regression with evolutionary computation,
- comparison of genetic algorithms with the PSO algorithm,
- assessment of the algorithm's impact to fill in missing values,
- proposing a novel method to select model weights using genetic algorithms instead of gradient algorithms,
- achieving high results - accuracy: 94.55% and F1-score: 93.56%.

The main disadvantages of the solution are:

- the model requires testing on a larger data set,
- in the case of larger data sets, the weight optimization process can be much more time consuming.
- perhaps other biology-inspired algorithms would have achieved better results

As part of our further work, we plan to use genetic algorithms to optimize the weights of the neural network and use deep learning to solve problems in the survival prediction of hepatocellular cancer.

7. Conclusion

This work focused on training the logistic regression classifier. The three possibilities to fuse logistic regression with genetic algorithms in the survival prediction problem of hepatocellular carcinoma are tested. In the first experiment, the model parameters were optimized. In the

second, a genetic selection of features was added, while the third experiment is a new approach to logistic regression model training - genetic selection of weights. Subsequent experiments allowed obtaining better and better results. The best results achieved had an accuracy of 94.55% and an f1-score of 93.56%. This shows how modifying a typical logistic regression allows for achieving significantly better results. In the future, we plan to work on the detection of hepatocellular carcinoma on larger data sets using the ensemble method and deep learning (especially neural networks) combining these methods with the genetic algorithms.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.compbiomed.2021.104431.

References

[1] Moloud Abdar, U Rajendra Acharya, Nizal Sarrafzadegan, Vladimir Makarenkov, NE-nu-SVC: a new nested ensemble clinical decision support system for effective diagnosis of coronary artery disease, in: IEEE Access 7, Nov. 2019, pp. 167605–167620.

[2] Moloud Abdar, Wojciech Ksiażek, U Rajendra Acharya, Ru-San Tan, Vladimir Makarenkov, Paweł Pławiak, A new machine learning technique for an accurate diagnosis of coronary artery disease, in: Computer methods and programs in biomedicine 179, Oct. 2019, p. 104992.

[3] Moloud Abdar, Vladimir Makarenkov, CWV-BANN-SVM Ensemble Learning Classifier for an Accurate Diagnosis of Breast Cancer, May 2019, p. 146.

[4] Moloud Abdar, Vivi Nur Wijayaningrum, Sadiq Hussain, R. Alizadehsani, Pawel Plawiak, U Rajendra Acharya, Vladimir Makarenkov, IAPSO- AIRS: a novel improved machine learning-based system for wart disease treatment, in: Journal of Medical Systems 43, 2019, pp. 1–23.

[5] Moloud Abdar, Neil Yuwen Yen, Jason Chi-Shu Hung, Improving the diagnosis of liver disease using multilayer perceptron neural network and boosted decision trees, in: Journal of Medical and Biological Engineering 38, 2018, pp. 953–965.

[6] U Rajendra Acharya, Steven Fernandes, Joel En Wei Koh, Edward Ciaccio, Mohd Fabell, U. Tanik, Venkatesan Rajinikanth, Yeong Chai, Automated detection of Alzheimer's disease using brain MRI images–A study with various feature extraction techniques, in: Journal of Medical Systems 43, Aug. 2019.

[7] Liaqat Ali, Iram Wajahat, Noorbakhsh Amiri Golilarz, Fazel Keshtkar, Syed Ahmad Chan Bukhari, LDA–GA–SVM: improved hepatocellular carcinoma prediction through dimensionality reduction and genetically optimized support vector machine, in: Neural Computing and Applications, 2020, pp. 1–10.

[8] Roohallah Alizadehsani, Mohamad Roshanzamir, Moloud Abdar, Adham Beykikhoshk, Abbas Khosravi, Saeid Nahavandi, Pawel Plawiak, Ru San Tan, U Rajendra Acharya, Hybrid genetic-discretized algorithm to handle data uncertainty in diagnosing stenosis of coronary arteries, in: Expert Systems, 2020.

[9] Panyanat Aonpong, Qing-qin Chen, Yutaro Iwamoto, Lanfen Lin, Hongjie Hu, Qiaowei Zhang, Yen-wei Chen, Comparison of Machine Learning-Based Radiomics Models for Early Recurrence Prediction of Hepatocellular Carcinoma, vol. 7, Dec. 2019, pp. 117–125.

[10] Zeinab Arabasadi, Roohallah Alizadehsani, Mohamad Roshanzamir, Hossein Moosaei, Ali Asghar Yarifard, Computer aided decision making for heart disease detection using hybrid neural network-Genetic algorithm, in: Computer methods and programs in biomedicine 141, Apr. 2017, pp. 19–26.

[11] Julius Balogh, David Victor, Emad Asham, Sherilyn Burroughs, Maha Boktour, Ashish Saharia, Xian Li, R. Ghobrial, Monsour Howard, Hepatocellular carcinoma: a review, in: Journal of Hepatocellular Carcinoma 3, Oct. 2016, pp. 41–53.

[12] Vitoantonio Bevilacqua, Antonio Brunetti, Gianpaolo Francesco Trotta, Leonarda Carnimeo, Francescomaria Marino, Vito Alberotanza, Arnaldo Scardapane, A deep learning approach for hepatocellular carcinoma grading, in: International Journal of Computer Vision and Image Processing, vol. 7, Apr. 2017, pp. 1–18, 2.

[13] Raluca Brehar, Delia Mitrea, Sergiu Nedevschi, Monica Platon Lupsor, Magda Rotaru, Badea Radu, Hepatocellular carcinoma recognition in ultrasound images using textural descriptors and classical machine learning, in: 2019 IEEE 15th International Conference on Intelligent Computer Communication and Processing (ICCP), 2019, pp. 491–497.

[14] Mingyu Chen, Bin Zhang, Win Topatana, Jiasheng Cao, Hepan Zhu, Sarun Juengpanich, Qijiang Mao, Yu Hong, Xiujun Cai, Classification and mutation prediction based on histopathology HE images in liver cancer using deep learning, in: NPJ Precision Oncology 4, 2020.

[15] Amita Das, U. Rajendra Acharya, Soumya Surath Panda, Sukanta Sabut, Deep learning based liver cancer detection using watershed transform and Gaussian mixture model techniques, in: Cognitive Systems Research 54, 2019, pp. 165–175.

[16] Etzioni Ruth, Nicole Urban, Ramsey Scott, Martin McIntosh, Stephen Schwartz, Brian Reid, Jerald Radich, Garnet Anderson, Leland Hartwell, Early detection: the case for early detection, in: Nature reviews. Cancer 3, May 2003, pp. 243–252.

[17] Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, Christian Gagné, DEAP: evolutionary algorithms made easy", in: Journal of Machine Learning Research 13, July 2012, pp. 2171–2175.

[18] Hammad Mohamed, Asmaa Maher, Kuanquan Wang, Feng Jiang, Moussa Amrani, Detection of abnormal heart conditions based on characteristics of ECG signals, in: Measurement 125, May 2018.

[19] Somaya Hashem, Mahmoud ElHefnawi, Shahira Habashy, Mohamed El-Adawy, Gamal Esmat, Wafaa Elakel, Ashraf Omar Abdelazziz, Mohamed Mahmoud Nabeel, Ahmed Hosni Abdelmaksoud, Tamer Mahmoud Elbaz, Hend Ibrahim Shousha, Machine learning prediction models for diagnosing hepatocellular carcinoma with HCV-related chronic liver disease, in: Computer methods and programs in biomedicine 196, May 2020, p. 105551.

[20] Mahbouba Hattab, Ahmed Maalel, and Henda Ben Ghezala. "Towards an oversampling method to improve hepatocellular carcinoma early prediction". In: Digital Health in Focus of Predictive, Preventive and Personalised Medicine Pp 139-148 International Conference on Digital Health Technologies (ICDHT 2019)At: Hammamet ( ).

[21] John Henry Holland, Genetic Algorithms and Adaptation, 1984, pp. 317–333.

[22] Ferdib Al-Islam, Laboni Akter, Md Milon Islam, Hepatocellular carcinoma patient's survival prediction using oversampling and machine learning techniques, in: 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), 2021. DHAKA, Bangladesh.

[23] Joanna Jaworek-Korjakowska, R. Tadeusiewicz, Assessment of dots and globules in dermoscopic color images as one of the 7-point check list criteria, in: 2013 IEEE International Conference on Image Processing, 2013, pp. 1456–1460.

[24] Rajesh N.V. P.S. Kandala, Ravindra Dhuli, Paweł Pławiak, Ganesh Naik, Hossein Moeinzadeh, Gaetano D. Gargiulo, Gunnam Gunnam, Towards Real-Time Heartbeat Classification: Evaluation of Nonlinear Morphological Features and Voting Method, Sensors 19 (2019) (Basel, Switzerland), MDPI.

[25] James Kennedy, Eberhart Russell, Particle Swarm Optimization, vol. 4, 1995, pp. 1942–1948, 4.

[26] Wojciech Książek, Moloud Abdar, U. Acharya, Paweł Pławiak, A novel machine learning approach for early detection of hepatocellular carcinoma patients", Cognitive Systems Research 54 (2019) 116–127.

[27] Wojciech Książek, Mohamed Hammad, Paweł Pławiak, U Rajendra Acharya, Ryszard Tadeusiewicz, Development of novel ensemble model using stacking learning and evolutionary computation techniques for automated hepatocellular carcinoma detection, Biocybernetics and Biomedical Engineering 40 (2020) 1512–1524, 4th ed., Elsevier.

[28] Bjoern H. Menze, Bernd Michael Kelm, Ralf Masuch, Uwe Himmelreich, Peter Bachert, Wolfgang Petrich, Fred A. Hamprecht, A comparison of Random Forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data, in: BMC Bioinformatics 10, 2009, p. 213, 1.

[29] Zbigniew Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, third ed., Springer-Verlag, Berlin, Heidelberg, 1996.

[30] Elham Nasarian, Moloud Abdar, Mohammad Amin Fahami, Roohallah Alizadehsani, Sadiq Hussain, Ehsan Basiri, Mariam Zomorodi, Xujuan Zhou, Paweł Pławiak, U Rajendra Acharya, Ru San Tan, and Nizal Sarrafzadegan. "Association between work-related features and coronary artery disease: a heterogeneous hybrid feature selection integrated with balancing approach", in: Pattern Recognition Letters 133, 2020, pp. 33–40.

[31] Akash Nayak, Esha Baidya Kayal, Manish Arya, Jayanth Culli, Sonal Krishan, Sumeet Agarwal, Mehndiratta Amit, Computer-aided diagnosis of cirrhosis and hepatocellular carcinoma using multi-phase abdomen CT, in: International journal of computer assisted radiology and surgery 14, Aug. 2019, pp. 1341–1352, https://doi.org/10.1007/s11548-019-01991-5, 8.

[32] Ruilin Pan, Tingsheng Yang, Jianhua Cao, Ku Lu, Zhanchao Zhang, Missing data imputation by K nearest neighbours based on grey relational structure and mutual information, in: Applied Intelligence 43, 2015, 3.

[33] Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Thirion Bertrand, Olivier Grisel, Mathieu Blondel, Prettenhofer Peter, Ron Weiss, Dubourg Vincent, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, Edouard Duchesnay, Scikit-learn: machine learning in Python, in: Journal of Machine Learning Research 12, 2011, pp. 2825–2830.

[34] Augustyniak Piotr, Ryszard Tadeusiewicz, Assessment of electrocar-diogram visual interpretation strategy based on scanpath analysis, in: Physiological measurement 27, 2006, pp. 597–608, 7.

[35] Paweł Pławiak, Novel genetic ensembles of classifiers applied to myocardium dysfunction recognition based on ECG signals, in: Swarm and Evolutionary Computation 39C, Apr. 2018, pp. 192–208.

[36] Paweł Pławiak, Moloud Abdar, U Rajendra Acharya, Application of new deep genetic cascade ensemble of SVM classifiers to predict the Australian credit scoring, in: Applied Soft Computing 84, Nov. 2019, p. 105740.

[37] Paweł Pławiak, Moloud Abdar, Joanna P. lawiak, Vladimir Makarenkov, U Rajendra Acharya, DGHNL: a new deep genetic hierarchical network of learners for prediction of credit scoring, in: Information Sciences 516, Apr. 2020, pp. 401–418.

[38] Paweł Pławiak, U Rajendra Acharya, Novel deep genetic ensemble of classifiers for arrhythmia detection using ECG signals, in: Neural Computing and Applications 32, Aug. 2020, pp. 11137–11161.

[39] Paweł Pławiak, Ryszard Tadeusiewicz, Approximation of phenol concentration using novel hybrid computational intelligence methods, in: International Journal of Applied Mathematics and Computer Science 24, 2014, pp. 165–181, 1.

[40] Leszek Rutkowski, Computational Intelligence: Methods and Techniques, Springer, 1992.

[41] Miriam Seoane Santos, Pedro Henriques Abreu, Pedro J. Garca-Laencina, Adlia Simo, Armando Carvalho, in: A New Cluster-Based Oversampling Method for Improving Survival Prediction of Hepatocellular Carcinoma Patients, 2015, p. 58.

[42] Miriam Santos, Pedro Henriques Abreu, Pedro García-Laencina, Adelia Simao, Armando Carvalho, A new cluster-based oversampling method for improving survival prediction of hepatocellular carcinoma patients, in: Journal of biomedical informatics 58, Oct. 2015, pp. 49–59.
[43] Masaya Sato, K. Morimoto, S. Kajihara, R. Tateishi, S. Shiina, K. Koike, Y. Yatomi, Machine-learning approach for the development of a novel predictive model for the diagnosis of hepatocellular carcinoma, in: Scientific Reports 9, 2019.
[44] Ramit Sawhney, Puneet Mathur, Ravi Shankar, in: A Fiery Algorithm Based Wrapper-Penalty Feature Selection Method for Cancer Diagnosis, 2018, pp. 438–449.
[45] Joanna Szaleniec, Maciej Wiatr, Maciej Szaleniec, SkłAdzień Jacek, Jerzy Tomik, Krzysztof Ole, Ryszard Tadeusiewicz, Artificial neural network modelling of the results of tympanoplasty in chronic suppurative otitis media patients, in: Comput. Biol. Med. 43, 2013, pp. 16–22, 1.
[46] Maciej Szaleniec, Ryszard Tadeusiewicz, Małgorzata Witko, How to select an optimal neural model of chemical reactivity?, in: Neurocomputing 72, 2008, pp. 241–256, 1.
[47] Turker Tuncer, Fatih Ertam, Neighborhood component analysis and reliefF based survival recognition methods for Hepatocellular carcinoma, in: Physica A: Statistical Mechanics and its Applications 540, Feb. 2020, p. 123143.
[48] Zi-Mei Zhang, Jiu-Xin Tan, Fang Wang, Fu-Ying Dao, Zhao-Yue Zhang, Hao Lin, Early diagnosis of hepatocellular carcinoma using machine learning method, in: Frontiers in Bioengineering and Biotechnology 8, 2020, p. 254.
[49] Mariam Zomorodi, Moloud Abdar, Zohreh Davarzani, Xujuan Zhou, Paweł Pławiak, U Rajendra Acharya, Hybrid particle swarm optimization for rule discovery in the diagnosis of coronary artery disease, Expert Systems 38 (2021) 1–17, 1st ed., Wiley.