Hybridization of Neural Networks and Sine Cosine Algorithm for an Optimal Neural Network Architecture Applied to Prevent Heart Attacks

https://doi.org/10.3991/ijoe.v18i05.29463

Maryem Hourri(✉), Noureddine Alaa
Cadi Ayyad University, Marrakech, Morocco
maryemhourri@gmail.com

Abstract—Artificial intelligence and deep learning provide very good results, if it is well adjusted. In this work, we will proceed to perform the deep learning results through the optimization of the neural network architecture. For this purpose, especially for supervised algorithms, Hybridization of neural networks and the sinus-cosine algorithm will perform classification problems. The role of this method is to escape the groping method in choosing the optimal architecture of neural networks. The goal of our method is to build an optimal neural network architecture, without falling into an over fitting problem. To demonstrate the effectiveness of our work, an application part with experimental results is included: with an application in medicine especially heart attack. The goal of our work is to develop an efficient hybrid classifier, using machine learning and sine cosine algorithm to detect heart attack and minimize the number of heart attack not predicted. Through this hybridization technique we should have a low error with satisfying classification results. This method of hybridization can be applied for different issues.

Keywords—neural networks, sine cosine, hybridization, classification, heart attack, deep learning

1 Introduction

Artificial intelligence has become a common way to solve complex problems in many fields. It allows the use of intelligent algorithms that can detect patterns, for a better classification like the one used in the book [1], that other programs or algorithms cannot find (the links between different parameters or inputs). Artificial intelligence or deep learning is based on the same principles as biological networks; the ultimate goal of IA is to enable computers or machines to be capable of thinking like human beings. This would be possible through the combination of: computer systems, data with management systems and advanced AI algorithms (code) like in the following work [12].

Considering its efficient results, artificial intelligence is currently used in several fields such as medicine as shown in this example [11], finance, security and many other services.

Currently, several research projects are performing hybridization in order to improve the learning capacity of the neural networks that constitute the algorithms of artificial intelligence.

The hybridization of deep learning algorithms and meta heuristic algorithms allows to have efficient results. Among these works we cite the article [13]. But the problem that generally arises is how to choose a good architecture, which is more optimal, in order to obtain good results.

In our case, we will precede to a hybridization of a deep learning algorithm and a meta heuristic method which is the sine cosine algorithm, the combination of these two algorithms optimizes the error and produces efficient results and good quality compared to other methods. This is proven by the study and analysis performed in this article.

And this by trying to have an optimal architecture that will allow the neural networks to extract the best possible results. The goal of these different researches is to improve the performances and the results to be able to solve in an optimal way the complex problems that we can meet in the real life.

To demonstrate the effectiveness of our work we will proceed to applications. The applications will be in medicine to prove that our work can be used and give good results. In medicine the application will be prediction of heart attack in order to be able to prevent it. Because of heart attack, thousands or even billions of patients are died every year, that's why we chose to make an application on this problem to help predict and classify this disease for a better detection.

This article is organized as follows: After a general introduction, we find section 2 which represents a review of the literature on neural networks, the basic sine cosine algorithm (SCA) and other standard classification methods by highlighting the programming of their algorithms. Section 3 presents the hybridization of SCA and neural networks and an application for better optimization. Section 4 shows the results of the hybridization of the two methods and the comparison of the different methods. Finally, the last section is the one reserved for the discussion of the different results and potential future work. Writing a new document with this template.

You may also simply delete all the text in this document, paste yours and format it with the styles.

2 Background and materials

In this section we will proceed to the definition of the different methods that we are going to use, and especially the methods with whom we will compare our model to demonstrate its efficiency.

2.1 Classification trees

The decision tree is a model used in data mining and also in machine learning algorithms. The decision tree consists of leaves which are exactly the desired values as a result and branches which are the input values that subsequently lead to the previous values. The decision tree is mainly related to the explanation of the data but not to the decisions themselves.

It is a predictive method of supervised learning since we know the data and the target value to build this decision tree. the method is well demonstrated in this following article [6]. The first step is to make a fractionation for choosing the best and most important attributes so that; we can create the different diversified groups using the gain and the entropy.

Algorithm 1 steps of the classification trees algorithm. We assume that the target variables have m distinct values (the class labels). For a node S; we compute its entropy with respect to the target,

Step 1: We split S into m groups: C1... Cm.

Step 2: Compute the probability pi of an element of S in Ci.

Step 3: Compute the entropy and the gain using the following formula to finally have our classification:

$$H(p_1, p_2, .., p_m) = \sum_{i=1}^{m} p_i \, log(p_i) \tag{1}$$

H(S): measures the deviation of the distribution of the target variable from the uniform distribution.

Step 4: Repeat the steps until H(S)=0, if S is homogeneous (all elements are in the same class).

Step 5: Display the decision tree.

2.2 Fuzzy logic algorithms

Fuzzy logic is part of Boolean logic, taking part of the mathematical theory of fuzzy parts, according to [7]. The characteristic of this method is the introduction of the notion of degree in the confirmation of a condition, giving a flexibility to the reasoning without being restricted in a binary value or true or false or other. This will allow to take into account the imprecision and the uncertainties.

The interest of using fuzzy logic is its ability to take part not on numerical variables, but-on linguistic variables: on qualitative variables (beautiful, ugly, very beautiful, fair, ...).

The added value is the ability to work on linguistic expressions. This method is very useful when we are dealing with systems that are almost impossible to model.

Classical logic is a part of mathematics in classical logic the interpretations are binary. That's why we will use fuzzy logic to have a decision that is both true and false at the same time, with a certain degree of belonging to each of these two beliefs.

In fuzzy logic, a fact no longer has a strict belonging to a belief, but a "fuzzy" belonging.

Algorithm 2 steps of the fuzzy logic algorithm

Step1: Let X = [x1, x2, ..., Xn] the set of data points and V = [v1,v2, ..., vc] the set of centers.

Step2: Select c cluster centers randomly.

Step3: Calculate the fuzzy membership µij using the formula:

$$\mu ij = \frac{1}{\sum_{k=1}^{c} (\frac{dij}{dik})^{\frac{2}{m-1}}} \quad (2)$$

Where,
'm' is the fuzziness index m € [1, ∞].
'c' represents the number of cluster center.
'μij' represents the membership of *ith* data to *jth* cluster center.
'dij' represents the Euclidean distance between *ith* data and *jth* cluster center.
Step4: Compute the fuzzy centers 'vj' using:

$$vj = \frac{\sum_{i=1}^{n} (\mu ij^{m}) xi}{\sum_{i=1}^{n} (\mu ij^{m})} \qquad \text{for } j=1\ldots c \quad (3)$$

'vj' represents the *jth* cluster center.
Step5: Repeat step2 and step3 until the minimum value is achieved.

2.3 Deep neural networks (ANNs)

An artificial neural network is made up of a set of connections that play the role of a signal transmitter to the different neurons. Each received signal is processed and then transmitted to other nodes connected to it. Each weight is the result of a connection between neurons which can have multiple input and output connections at the same time. The output of this signal is calculated using a non-linear function called the activation function, as specified in the following work [6].

Among the different activation functions that can be found we quotes: Historically, the sigmoid function is the oldest and most popular activating function. It is defined as:

$$S(t) = \frac{1}{1+e^{-t}} \quad (4)$$

The ultimate role of the softmax activation is to normalize the probability distribution made up of K proportional probabilities that's why we use it:

$$f(xj) = \frac{e^{xj}}{\sum_{i=1}^{n} e^{xi}} \qquad \text{for } j=1...n \quad (5)$$

Before being able to use neural networks, understanding their characteristics is essential. The choice of which model to use depends largely on the data as well as the application, but only training the data and the selection of the setting will require a lot of experimentation. The appropriate choice of the model as well as the fitness function and the learning algorithm will give birth to a powerful neural network.

Patternet is a function of deep neural network, which constitutes a network of neurons with a well-defined number of hidden layers (hiddenSizes), a learning function (trainFcn), and finally a performance function (performFcn). It is represented by net = patternnet (hiddenSizes, trainFcn, performFcn).

The goal of this function is to be able to classify the inputs to arrive at well-defined classes by specifying the architecture of the neural network and generating a pattern recognition network.

Patternet is a deep learning function suitable for Matlab which helps in solving complex classification and prediction problems.

The algorithm that allows us to generate the Patternet is below:

Algorithm 3 steps of the patternet algorithm

Step1: Input the data: x: the attributes and t: the target.

Step2: Choose the architecture of the network: $[a1, a2, \dots, aq]$.

(aj: represent the number of neurons in the jth hidden layer)

Step3: Construction of the architecture of the classification model: net = patternnet([a1, a2,...,aq])

Step4: Train the model: net = train (net, x, t)

Step5: Generation of the simulate target y = net(x).

2.4 Sine cosine algorithm

Sine Cosine SCA is a meta-heuristic algorithm developed through sine cosine mathematical functions, in order to be able to use it in the resolution of optimization problems. This optimization method was first designed in Mirjalili in 2016 according to the following reference [4]. The modification of each search agent using the sine cosine algorithm is performed through the following two equations:

$$X_i^{t+1} = \begin{cases} X_i^t + r_1 \times sin(r_2) \times |r_3 P_i^t - X_i^t|, r_4 < 0.5 \\ X_i^t + r_1 \times cos(r_2) \times |r_3 P_i^t - X_i^t|, r_4 \geq 0.5 \end{cases} \quad (6)$$

With r2 : random variable such that r2 in $[0,2\pi]$, r3 a random variable, r4 is used for the choice of search paths, sine or cosine, based on the random values of the first equation, Pi: the objective solution and r1 is between m and 0 in order to balance the iteration and represented below: $r_1(\mathrm{i}) = \mathrm{m} \times (1 - \frac{\mathrm{i}}{\mathrm{imax}})$ where:

- imax : The total number of iteration
- m >0
- i: the present iteration

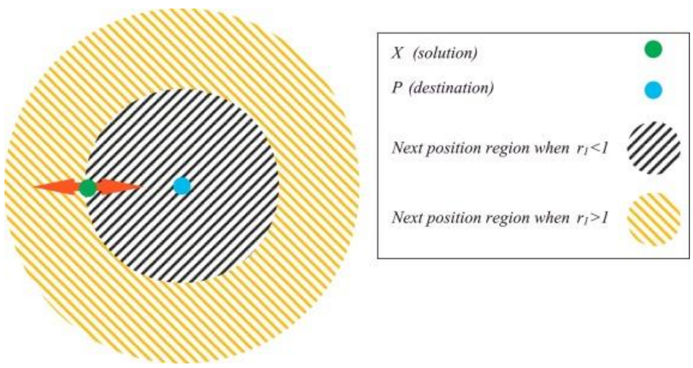

Fig. 1. Basic sine cosine algorithm

SCA is an optimization algorithm which refers to a population already generated by sine and cosine mathematical functions. Among the similar algorithms we find MHA, SCA which generate start a set of random solutions. Subsequently, the different solutions are evaluated due to the objective function. After this step the different resolutions are evaluated, we choose the best among them, in order to store the optimal solution and represent it as the next destination point. The different results are updated to generate new ones, through the sine and cosine functions. The algorithm ends when the number of iterations of the algorithm is satisfied.

The algorithm 4 demonstrates the different stages of this process

Algorithm 3 steps of sine cosine algorithm:

Step1: Initialize a (solutions) (Y).

Step2: Evaluate each solution by the objective function.

Step3: Update the best solution obtained.

Step4: Update the parameters r1, r2, r3, and r4.

Step5: Update the solution using the equation 1 defined before until reaching the maximum number of iterations.

Step6: Return the optimal solution.

The goal of the basic SCA is to be able to find new promising spaces of solutions in a rather short time frame. In this paper we will proceed to a hybridization in order not to fall on the local optima and at the same time to improve the classification or the prediction of the problems by optimizing the objective function. In our case the objective function is the fitness function. The fitness function F is defined by:

$$\mathrm{F}\left([a_1, a_2, \ldots, a_q]\right) = \frac{1}{\mathrm{N}} \sum_{\mathrm{i}=1}^{\mathrm{n}} (y_i - t_i)^2 \quad (7)$$

y_i: predicted output by the algorithm 2

t_i: the desired target

N: the number of individuals

a_j: Represent the number of neurons in the jth hidden layer.

3 Proposed new algorithm: (DeepSCA)

The SCA method is a meta-heuristic algorithm which is based on optimization in large solution domain. On the other hand, in certain cases its capacity to escape the local optimum is weakened, mainly due to the lack of diversity and the extraction of the best available memory, in terms of quality of the solution according to [8] [7]. This is why we are going to proceed with a hybridization method in order to be able to improve the results obtained.

Algorithm 5 steps of the hybrid algorithm

Step1: Initialize the neural network with the algorithm 3.

Step2: Create a neural network.

Step3: Train the network net using the training data.

Step4: Estimate the targets using the trained network y = net(x),
y: the output generated by the patternnet.

Step5: Calculate the fitness function with the formula (2.7)

Step6: Run the sine cosine algorithm, with the algorithm 4.
Step7: Get the fitness optimized.
Step8: Get the optimal architecture of neural networks.

4 Application

The detection of hearth attack is quite difficult by human analysis in some cases due to the large number of data and samples. That's why we should resort to an automatic method. But to have a good prediction it is not enough to make it just automatic but also efficient. The analysis of the detection will be based on the information's already registered.

In the following we will display the data that we will use as an application, and then the different results obtained will be represented in figures and finally a comparison of the different methods to demonstrate the efficiency of our model.

4.1 Heart attack analysis classification

The data used in our application as well as all the information concerning the attributes used are taken from [14].

The database contains 76 attributes, but all published experiments refer to using a subset of 14 of them. In particular, the Cleveland database is the only one that has been used by ML researchers to this date. The "goal" field refers to the presence of heart disease in the patient. It is integer valued from 0 (no presence) to 4. Experiments with the Cleveland database have concentrated on simply attempting to distinguish presence (values 1,2,3,4) from absence (value 0).

The data contains 300 observations with 13 categorial/symbolic attributes and one target:

Age: age in years
Sex: sex (1 = male; 0 = female)
Cp: chest pain type

- Value 1: typical angina
- Value 2: atypical angina
- Value 3: non-anginal pain
- Value 4: asymptomatic

Trestbps: resting blood pressure (in mm Hg on admission to the hospital)
Chol: serum cholestoral in mg/dl
Fbs: (fasting blood sugar > 120 mg/dl) (1 = true; 0 = false)
Restecg: resting electrocardiographic results

- Value 0: normal
- Value 1: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV)
- Value 2: showing probable or definite left ventricular hypertrophy by Estes' criteria

Thalach: maximum heart rate achieved
Exang: exercise induced angina (1 = yes; 0 = no)
Oldpeak = ST depression induced by exercise relative to rest
Slope: the slope of the peak exercise ST segment

- Value 1: upsloping
- Value 2: flat
- Value 3: downsloping

Ca: number of major vessels (0-3) colored by flourosopy
Thal: 3 = normal; 6 = fixed defect; 7 = reversable defect
Attribute: Num: diagnosis of heart disease (angiographic disease status)

- Value 0: < 50% diameter narrowing
- Value 1: > 50% diameter narrowing
(in any major vessel: attributes 59 through 68 are vessels)

The purpose of this study was to predict heart attack status for each patient. We performed an 80% - 10% data distribution to create independent training and test sets. In the training set, we also isolated 10% of patient to create an independent validation set. The splits were performed in a satisfied manner to maintain the same proportion of default cases in the training, validation and test sets.

To prove the utility and the performance of our model it is necessary to compare it with other methods (Table 1 and 2).

The purpose of this classification is to know which people are susceptible to have a heart attack.

4.2 Result and discussion

By applying the classical method which is the basic neural network with a random choice of number of neuron, in this case we used a network of 3 hidden layers: the input layer contains 4 neurons, the first hidden layer contains 1 neurons, the second one 9 neurons, the third one 19 neurons and the last layer which is the output layer contains 2 neurons.

As a result, we have the confusion matrix in Figure. 2 with a 55% of good predictions: classification that are correctly classified and 45% are bad classified.

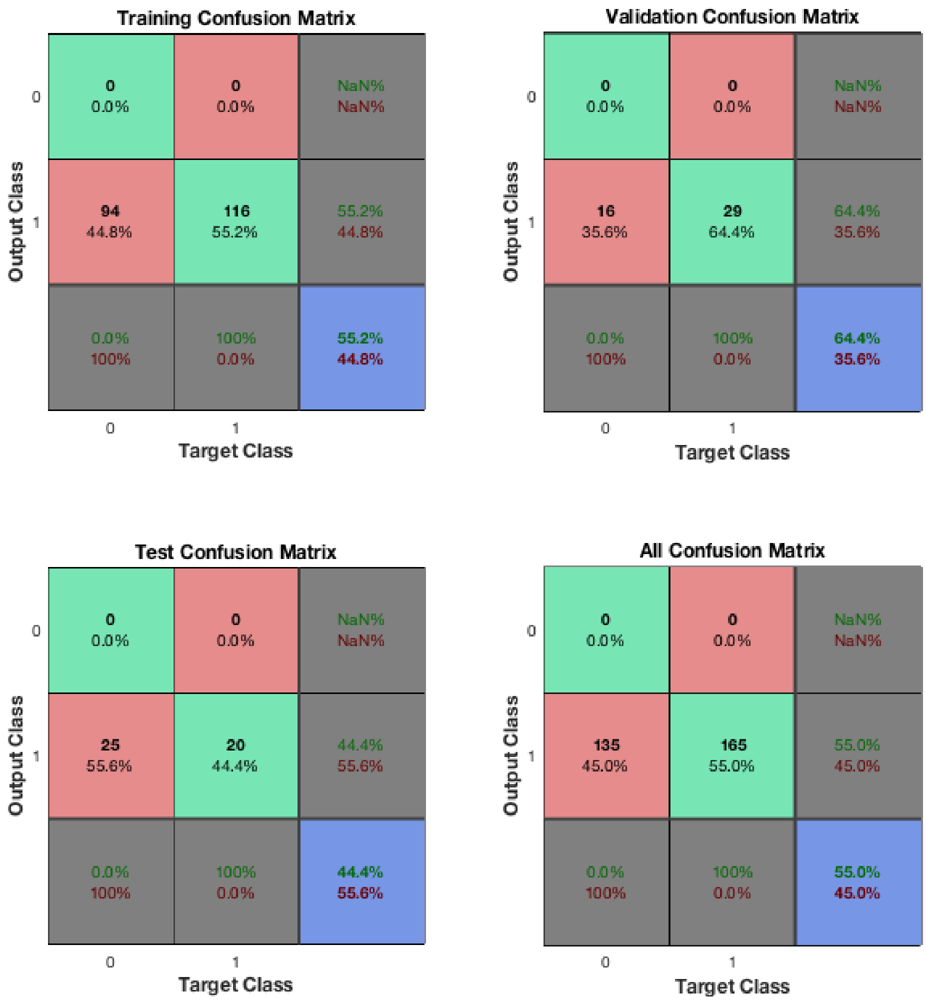

Fig. 2. Confusion matrix obtained by the deep neural network algorithm: Heart attack

Finally, by applying our proposed model: Hybridization of neural networks and Sine Cosine Algorithm for better classification we obtain satisfactory results (Figure 3). As a result, we have the confusion matrix in Figure 3 with a 91.4% of the good prediction: classification that are correctly classified and 8.6% are bad classified.

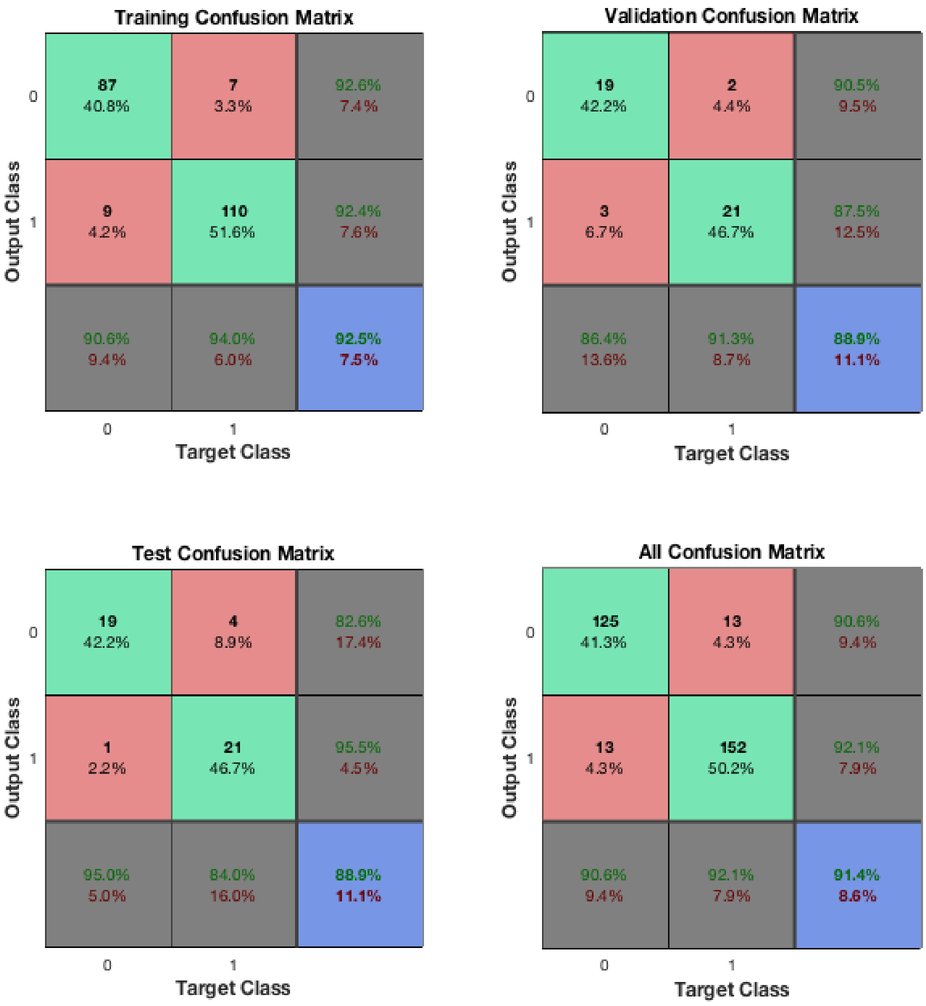

Fig. 3. Confusion matrix obtained by our algorithm: Hearth attack

The optimal architecture generated by our algorithm allow us to have a good classification that is represented in Figure 4.

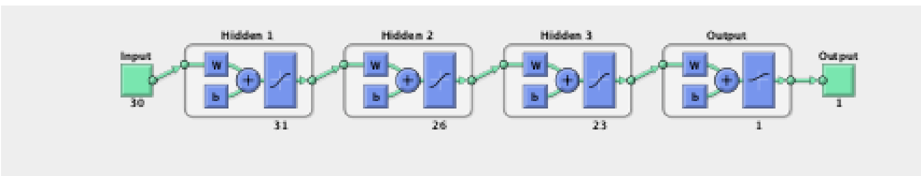

Fig. 4. Optimal architecture

Our work has shown that a simple deep learning algorithm with randomly placed numbers of neurons is not efficient enough, and the choice of several combinations of each number of neurons of each hidden layer to have the best performance is a waste of time and a risk of not trying all possible combinations. While our model allows to obtain the best combination of neural networks for each hidden layer for better classification.

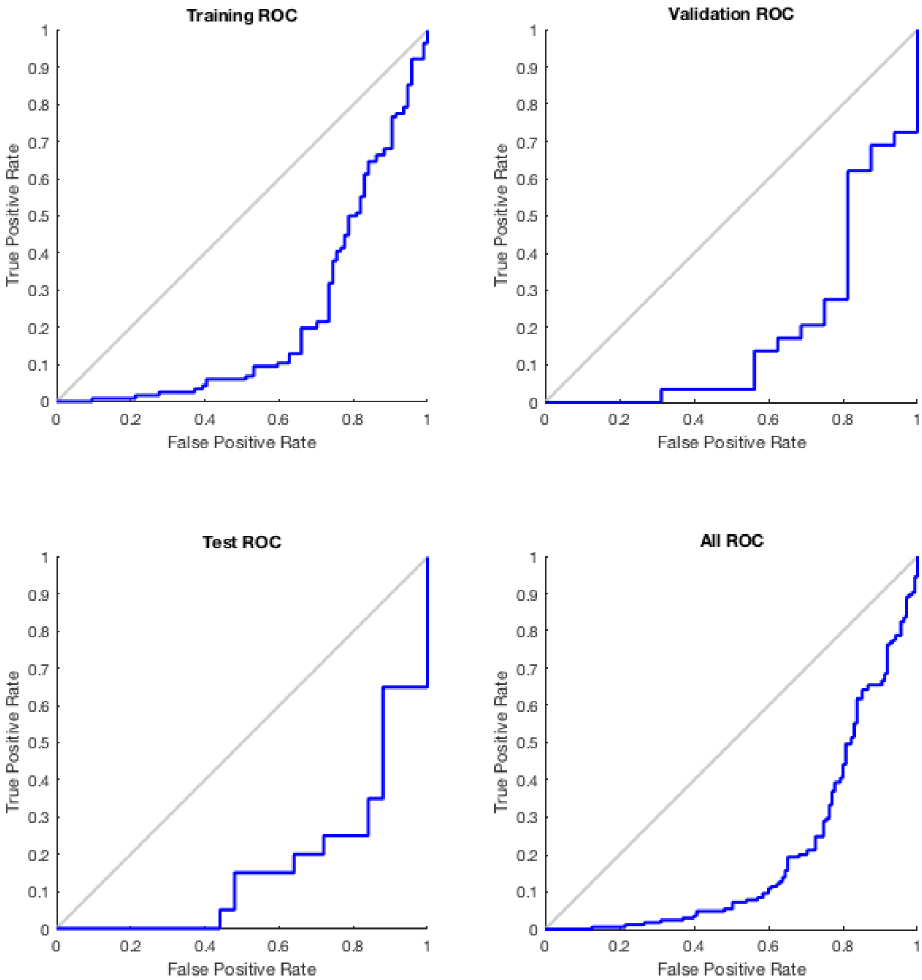

Fig. 5. ROC curve obtained by the deep neural network algorithm: Heart attack

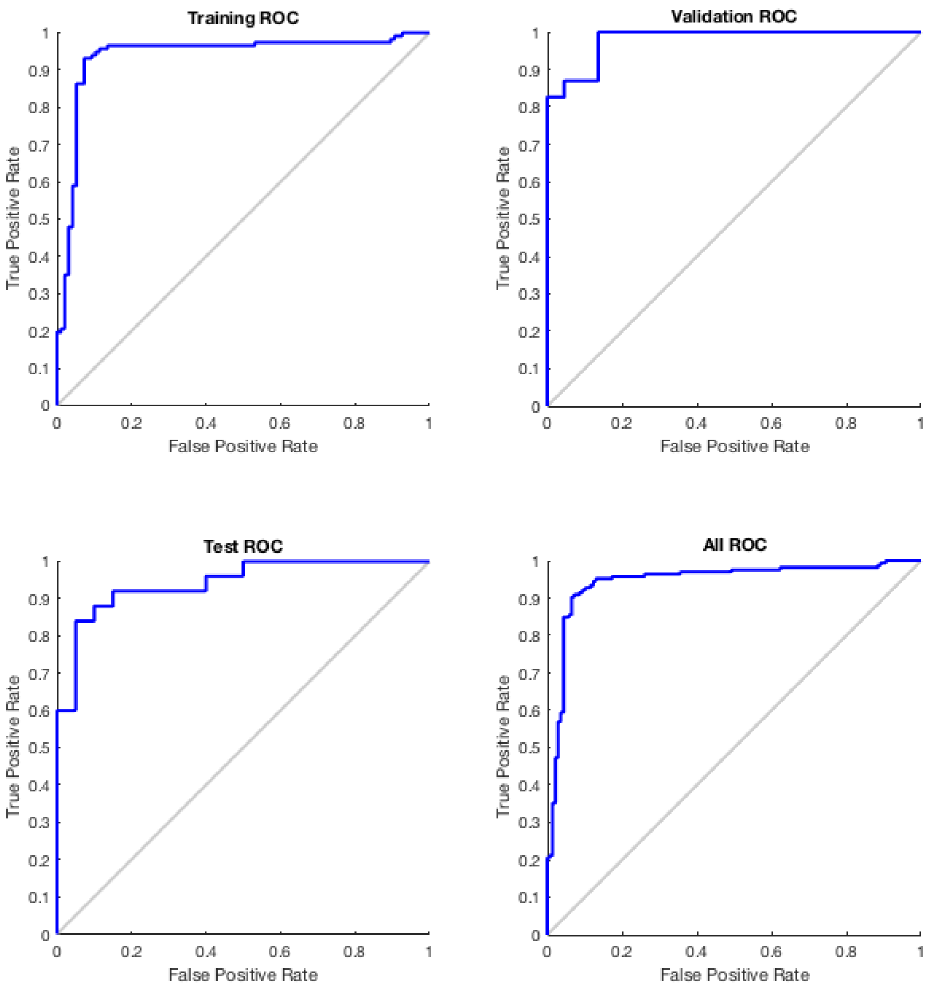

Fig. 6. ROC curve obtained by our algorithm: Heart attack

For this first comparison between the result of the Deep neural network and our hybrid model we will need the following relations:

The accuracy of a test means the ability to differentiate the patient susceptible to heart attack or not correctly.

$$Accuracy = (TP + TN) / (TP + FN + FP + TN) \tag{8}$$

The sensitivity of a test means the ability to determine the patient susceptible to heart attack cases correctly. To estimate it, we should calculate the proportion of true positive in-patient cases.

$$Sensitivity = TP / TP + FN \tag{9}$$

The specificity of a test means the ability to determine the patient not susceptible to heart attack cases correctly. To estimate it, we should calculate the proportion of true negative in transaction cases.

$$Specificity = TN/(FP + TN) \quad (10)$$

Where TP indicates true positive, TN indicates true negative, FN indicates false negative, and FP indicates false positive.

Table 1. Comparison Deep neural netwoork algorithm and DeepSCA

Method	Accuracy	Sensitivity	Specificity
Classic algorithm	55	0	100
DeepSCA	90.1	90.6	92.1

Increasing the number of classifications in this case will allow us to prevent heart attack. Through all the results obtained and the comparison with existing methods, we see that our model is efficient and gives good results. This will allow in this case to detect heart attack eventually, to reduce the loss of time in the hospital area with a good prediction.

Table 2. Comparison of the different methods

Methods	Accuracy
Neural networks	0.55
Classification tree	0.56
Fuzzy Logic	0.12
Hybrid sine cosine and neural networks	0.914

Increasing the number of classifications in this case will allow us to prevent more heart attacks.

Through all these models we could prove that with the use of our hybridization algorithm, we can have an optimal architecture of the neural network without having to adjust the number of neurons in each layer to have a good classification with a high accuracy.

5 Discussion and futures works

In this study, the databases were reduced to accommodate the available GPU (8 GB). If the GPU memory is large, the following studies will be done on larger databases. Working on large databases will affect the quality of classification.

Finally, our work proves that the hybridization of the two algorithms deep learning algorithm and sine cosine algorithm optimizes our neural network architecture, in order to optimize the error and have a good classification. Therefore, the hybridization method between neural networks and Sine Cosine Algorithm improves the classification for better detection of people at risk of credit risk depending on the application.

Our approach can help banking systems detect people at risk in order to increase the number of accepted credit files and reduce rejected files. This method can be applied to other financial or other issues.

As future research, there are many works to do. To name a few an optimization using Particle Swarm Optimization, Grey Wolf Optimizer, Ant Lion Optimizer, Moth Flame Optimizer.

6 References

[1] Rawat, V., Suryakan (2019). "A Classification System for Diabetic Patients with Machine Learning Techniques". International Journal of Mathematical, Engineering and Management Sciences, 4(3), 729-744. https://doi.org/10.33889/IJMEMS.2019.4.3-057

[2] M. Abd Elaziz, D. Oliva, S. Xiong " An improved Opposition-Based Sine Cosine Algorithm for global optimization", Expert Systems with Applications Volume 90, Pages 484-500, 30 December 2017. https://doi.org/10.1016/j.eswa.2017.07.043

[3] Shubham Gupta, Kusum Deep, Seyedali Mirjalili, Joong Hoon Kim, "A modified Sine Cosine Al- gorithm with novel transition parameter and mutation operator for global optimization ", Expert Systems With Applications, Volume 154, 15 September 2020, 113395. https://doi.org/10.1016/j.eswa.2020.113395

[4] Mirjalili, "S. SCA: A sine cosine algorithm for solving optimization problems". Knowl. Based Syst. , 96,120–133, 2016. https://doi.org/10.1016/j.knosys.2015.12.022

[5] Somnath Maity, Sankar Kumar Roy, "A New Approach for Solving Type-2-Fuzzy Transportation Problem", International Journal of Mathematical, Engineering and Management Sciences, 4(3), 683- 696. https://doi.org/10.33889//IJMEMS.2019.4.3-054

[6] Mudunuru, V. R., Skrzypek, L. A. (2020). " A Comparison of Artificial Neural Network and Decision Trees with Logistic Regression as Classification Models for Breast Cancer Survival". International Journal of Mathematical, Engineering and Management Sciences, 5(6), 1170-1190. https://doi.org/10.33889/IJMEMS.2020.5.6.089

[7] H.Nenavath, D. Ravi Kumar Jatoth, D. wagatam Das"A synergy of the sine-cosine algorithm and particle swarm optimizer for improved global optimization and object tracking", Swarm and Evolu- tionary Computation, Volume 43, Pages 1-30, December 2018. https://doi.org/10.1016/j.swevo.2018.02.011

[8] S. Gupta,Kusum Deep "Improved sine cosine algorithm with crossover scheme for global optimiza- tion" , knowledge-Based Systems Volume 165, 2019, Pages 374-406, 2019. https://doi.org/10.1016/j.knosys.2018.12.008

[9] G.P. Zhang, " Neural Networks for Classification: A Survey", IEEE Trans-actions On Systems, Man, And Cybernetics-part C: Applications And Reviews, vol. 30, N. 4, 1958. https://doi.org/10.1109/5326.897072

[10] Gabli, Mohammed; Jaara, El Miloud; Mermri, El Bekkaye; " A Genetic Algorithm Approach for an Equitable Treatment of Objective Functions in Multi-objective Optimization Problems", IAENG International Journal of Computer Science, Vol. 41, Issue2 pp. 22-31, 2014.

[11] Shweikeh, E., Lu, J., & Al-Rajab, M. (2021). Detection of Cancer in Medical Images using Deep Learning. International Journal of Online and Biomedical Engineering (iJOE), 17(14), pp. 164–171. https://doi.org/10.3991/ijoe.v17i14.27349

[12] Lakshmanarao, A., Raja Babu, M., & Srinivasa Ravi Kiran, T. (2021). An Efficient Covid19 Epidemic Analysis and Prediction Model Using Machine Learning Algorithms. International Journal of Online and Biomedical Engineering (iJOE), 17(11), pp. 176–184. https://doi.org/10.3991/ijoe.v17i11.25209

[13] Chang, Y., & Abu-Amara, F. (2021). An Efficient Hybrid Classifier for Cancer Detection. International Journal of Online and Biomedical Engineering (iJOE), 17(03), pp. 76–97. https://doi.org/10.3991/ijoe.v17i03.19683

[14] https://archive.ics.uci.edu/ml/datasets/heart+disease

7 Authors

Maryem Hourri obtained her state engineering degree in finance and actuarial science in 2015, then she obtained a master's degree in auditing and management control at the National School of Business and Management in 2017, and is currently pursuing a doctorate in the Department of Applied Mathematics and Computer Science since 2018 in University Cadi Ayyad, Morocco (email: maryemhourri@gmail.com).

NourEddine Alaa received his Master of Science and PhD from the University of NancyI France respectively (1986) and (1989). He received the HDR in Applied Mathematics from the University of Cadi Ayyad, Morocco (1996). He is currently Professor of Modeling and Scientific Computing at the Faculty of Sciences and Technology of Marrakech. His research is geared towards nonlinear mathematical models and their analysis, digital processing applications and Intelligence Artificial (email: n.alaa@uca.ac.ma).

Article submitted 2022-01-12. Resubmitted 2022-02-26. Final acceptance 2022-02-26. Final version published as submitted by the authors.