Journal of Medical Systems (2019) 43: 220
https://doi.org/10.1007/s10916-019-1343-0

SYSTEMS-LEVEL QUALITY IMPROVEMENT

IAPSO-AIRS: A novel improved machine learning-based system for wart disease treatment

Moloud Abdar[1] · Vivi Nur Wijayaningrum[2] · Sadiq Hussain[3] · Roohallah Alizadehsani[4] · Pawel Plawiak[5] · U. Rajendra Acharya[6,7,8] · Vladimir Makarenkov[1]

Received: 21 March 2019 / Accepted: 13 May 2019 / Published online: 7 June 2019

Abstract

Wart disease (WD) is a skin illness on the human body which is caused by the human papillomavirus (HPV). This study mainly concentrates on common and plantar warts. There are various treatment methods for this disease, including the popular immunotherapy and cryotherapy methods. Manual evaluation of the WD treatment response is challenging. Furthermore, traditional machine learning methods are not robust enough in WD classification as they cannot deal effectively with small number of attributes. This study proposes a new evolutionary-based computer-aided diagnosis (CAD) system using machine learning to classify the WD treatment response. The main architecture of our CAD system is based on the combination of improved adaptive particle swarm optimization (IAPSO) algorithm and artificial immune recognition system (AIRS). The cross-validation protocol was applied to test our machine learning-based classification system, including five different partition protocols (*K*2, *K*3, *K*4, *K*5 and *K*10). Our database consisted of 180 records taken from immunotherapy and cryotherapy databases. The best results were obtained using the *K*10 protocol that provided the precision, recall, F-measure and accuracy values of **0.8908, 0.8943, 0.8916** and **90%**, respectively. Our IAPSO system showed the reliability of **98.68%.** It was implemented in Java, while integrated development environment (IDE) was implemented using NetBeans. Our encouraging results suggest that the proposed IAPSO-AIRS system can be employed for the WD management in clinical environment.

Keywords Wart disease · Data mining · Machine learning · Computer-aided diagnosis system · Artificial immune recognition system · Improved adaptive particle swarm optimization

This article is part of the Topical Collection on *Systems-Level Quality Improvement*

✉ Moloud Abdar
m.abdar1987@gmail.com

Vivi Nur Wijayaningrum
vivinurw@gmail.com

Sadiq Hussain
sadiq@dibru.ac.in

Roohallah Alizadehsani
ralizadehsani@deakin.edu.au

Pawel Plawiak
plawiak@pk.edu.pl

U. Rajendra Acharya
aru@np.edu.sg

Vladimir Makarenkov
makarenkov.vladimir@uqam.ca

[1] Département d'Informatique, Université du Québec à Montréal, Montréal, QC, Canada

[2] Department of Information Technology, Politeknik Negeri Malang, Malang, East Java, Indonesia

[3] Dibrugarh University, Dibrugarh, Assam, India

[4] Institute for Intelligent Systems Research and Innovations (IISRI), Deakin University, Geelong, Victoria, Australia

[5] Institute of Telecomputing, Faculty of Physics, Mathematics and Computer Science, Cracow University of Technology, Krakow, Poland

[6] Department of Electronics and Computer Engineering, Ngee Ann Polytechnic, Singapore, Singapore

[7] Department of Biomedical Engineering, School of Science and Technology, Singapore University of Social Sciences, Singapore, Singapore

[8] Department of Biomedical Engineering, Faculty of Engineering, University of Malaya, Kuala Lumpur, Malaysia

Introduction

Wart disease (WD) is a skin illness on the human body caused by the human papillomavirus (HPV). Three main groups of warts are: epidermodysplasia verruciform is (EV), cutaneous and mucosal [1]. The cutaneous warts are very general ailments in both adults and children which have significant prevalence with 3% to 13% among general population and approximately 33% among primary school children [2, 3]. There are *four* different types of warts such as common, flat, filiform, and plantar warts. Usually, common warts can be observed on the hands, flat warts are frequently observed on the legs and on the back of the hands [4], while plantar warts are usually seen on the soles of the feet [5]. This study mainly concentrates on common and plantar warts. There are various treatment methods for these types of wart diseases. However, this study considers two widely popular treatment methods: immunotherapy and cryotherapy. Since patients respond differently to different treatment methods, the physicians are interested to find the customized treatments. Therefore, it is important to characterize the wart disease, and thereby evaluate the therapy or treatment response on the patients having this disease. This study tries to provide a novel methodology based on machine learning (ML) techniques to classify the WD treatment response. There have been a few studies in the literature that have investigated the WD treatment response problem using different ML methods [1, 6, 7]. These studies suffer from the following drawbacks: first, the current ML-methods applied for WD evaluation have low accuracies and not robust enough to be reliable. Second, the current methods are not fully automated and as a result, the manual evaluation of the WD treatment response suffers from subjectivity and laboriousness. Furthermore, the current methods are biased, since the sample size is small. Hence, no extensive research has been done to explore all facets of ML-based strategies to investigate WD treatment response.

Different types of ML-based methods can be used in computer-aided diagnosis (CAD) system which has shown to be effective in providing second opinion to the physicians in many fields of healthcare [8] such as neurology [9, 10], cardiology [11, 12], endocrinology [13, 14], dermatology [15] and diabetic retinopathy [16]. This semi-automated detection approach not only save the lives of the patients but is also more economical and represents a faster diagnosis process.

It may be noted that due to the importance of ML techniques, it has been successfully applied on various diseases [16–31]. Akben [1] proposed a new methodology using the fuzzy informative images and decision trees for the detection of WD. Author has applied ML methods on both cryotherapy and immunotherapy data sets and obtained accuracies of 94.40% and 90% for cryotherapy and immunotherapy data sets, respectively. In another study by Uzun et al. [6] have used k-Nearest Neighbors (KNN) algorithm on cryotherapy and immunotherapy data sets. They have obtained the highest accuracy of 80% for both data sets for $K = 7$. Khozeimeh et al. [7] proposed an expert system for selection of wart treatment methods. In their study, they treated 90 patients with immunotherapy and 90 patients with cryotherapy and liquid nitrogen. They were able to predict the results of the treatment method with the accuracy of 80% using a fuzzy rule-based system.

Recently, particle swarm optimization (PSO) is one of the more prominent evolutionary computation methods introduced which offers several advantages such as [32]: (1) it is an intelligence-based paradigm, (2) does not have mutation calculation and overlapping issues, and (3) offers simplicity in calculation. The only challenge is that the PSO was initially non-adaptive in nature.

Adaptive approaches are well-known techniques used to improve the performance of conventional methods. These approaches have been effectively combined with different methods such as PSO [33] and genetic algorithms (GAs) [34]. The PSO method includes a collection of several particles that move in the search space in which each particle has *two* characteristics: position and velocity. Generally, the PSO method updates the position of particles using a personal best position (*pbest*) and a global best (*gbest*). The adaptive approaches allow a better features search capability over classical PSO [33]. The adaptive PSO further suffers from updating the velocity and position of particles, which lowers the performance of the CAD system. We have removed this weakness by setting the maximum and minimum velocity limit of particles and dynamically adjusting the control parameters c_1 and c_2 during the solution search process. We call this as improved adaptive particle swarm optimization (IAPSO) method.

The PSO algorithm has been extensively used for the diagnosis of heart disease [35, 36], in health care [37], for the Parkinson disease detection [38], for the detection of hemorrhage [39], in neurological application [40], for the breast cancer detection [41, 42] and for Alzheimer's disease detection [43, 44]. The adaptive PSO has also been used to attain better accuracy during the breast cancer diagnosis [45]. The effectiveness of APSO, which is an optimization method, has been shown in a variety of applications such as sonar image sequences [46], humanoid robots [47] and tandem blade optimization [48]. The velocity and position of particles have significant effect on the performance of APSO. The reason for that is that the position and velocity of each particle is updated during the iteration so that each particle moves closer to a better solution. In this study, we update the position and velocity of each particle to improve the performance of APSO. The proposed method is called IAPSO (improved APSO).

Our database consisted of 180 records taken from the original immunotherapy and cryotherapy data equally (90 records for each of the original data types). To avoid the effect of variation in the IAPSO method, we first normalized the data set as part of the data preparation and preprocessing step. Cross-validation (CV) protocol was employed to test our machine learning paradigm consisting of *five* partition protocols (*K*2, *K*3, *K*4, *K*5, and *K*10). The CV protocol was evaluated using several performance metrics that includes: precision, recall, F-measure, and accuracy. We benchmarked our algorithm against the previously published method called artificial immune recognition system (AIRS). We further evaluated our CAD system by swapping the core classifier of IAPSO (with AIRS) by *six* different classifiers (shown as a general block in Fig. 1) namely simple AIRS, Bayes network (BN), Multilayer Perceptron (MLP), J48, Random forest (RF), and hierarchical LVQ (H-LVQ) to understand the generalization behavior of the ML system. As part of the performance evaluation of the ML system, we further computed the reliability index (RI). The system was implemented in the Java, while integrated development environment (IDE) was implemented using NetBeans.

The velocity and position of particles in APSO was updated in *two* major steps. We first set the maximum and minimum velocity limits of particles. Then, the values of c_1 and c_2 were adjusted. Therefore, the main contributions of this study can be listed as follows: (1) it presents a new standard WD database, (2) it applies different partition protocols from each method to check the performance of the algorithms, (3) it improves the performance of classical PSO (IAPSO), and (4) it describes a novel evolutionary-based approach to improve the performance of AIRS using IAPSO for WD treatment response. Our results demonstrate that IAPSO can improve the performance of the AIRS method on a WD database. Moreover, the proposed methodology outperformed the five classical methods used for comparison.

The rest of the work is structured into *seven* main sections which are as follows. Section "Clinical data" describes the clinical data sets used in our research followed by the proposed methodology in Section "The proposed methodology". Section "IAPSO architecture" provides our proposed methodology for classification of the WD treatment response. The experimental protocols are presented in Section "Experimental protocols", followed by the results description in Section "Results". The performance evaluation and discussion are provided in Section "Performance evaluation", followed by Section "Discussion". Finally, we have presented the conclusions in Section "Conclusion".

Clinical data

The main data used in this paper comprised of two data sets: plantar and common warts data. Both of these data sets were gathered in the dermatology clinic of Ghaem Hospital in Mashhad, IRAN between the period of January 2013 and February 2015. These data sets can be downloaded from the UCI machine learning repository [7, 49–51]. The information from 180 patients with plantar and common warts was recorded. Each data set includes 90 records. We believed that the numbers of records in these data sets were not enough for optimal performance of the new algorithm. To tackle this challenge, we introduced a new standard WD database which was designed by combining both data sets and therefore consisted of 8 features selected for 180 records. Thus, the new data set: (1) includes more records; and (2) has more features, compared to the old data sets. The important point about the combined data set is that it has no missing values, which is very critical for ML techniques. More information about the attributes and the statistical summary of the original and new data sets are briefly discussed below.

The cryotherapy and immunotherapy data sets

The first data set includes *seven* features from 90 patients, collected when the cryotherapy approach was used. The second data set includes *eight* factors gathered from 90 patients when the immunotherapy approach was used. More information about these two data sets can be found in [7, 49–51].

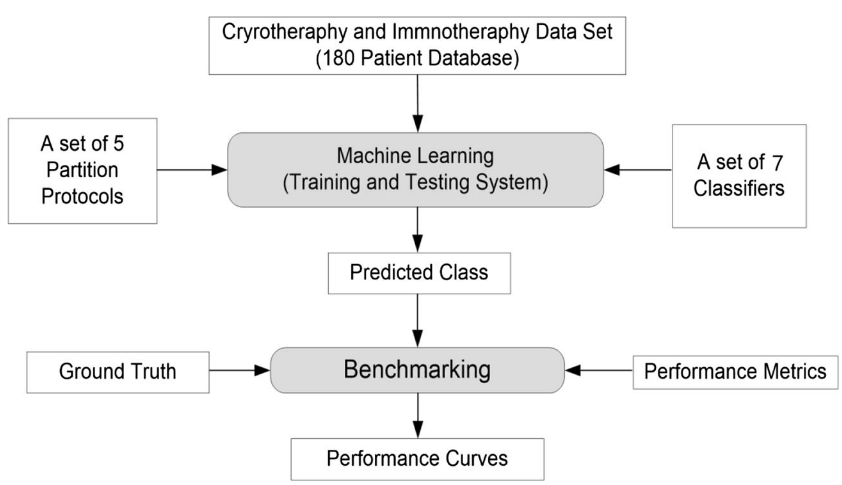


Fig. 1 Global architecture of application of machine learning algorithms for Wart disease prediction

The combined wart data (CWD)

The original data sets used *two* different treatment methods. Thus, we combined both data sets to make a new data set. In addition, we would argue that these treatment methods can be considered as a prediction feature for WD. Since the first data set includes *seven* factors and the second data set includes *eight* factors, we tried to use all common factors available in both data sets. Hence, the induration diameter of initial test feature existing in the second data set was removed. Furthermore, we added a new factor named "Procedure of treatment" as an input variable with other common factors (see Table 1):

The proposed methodology

This study applied a new machine learning (ML) system on the WD data set. The proposed ML-based system (named IAPSO for AIRS) is presented in Fig. 2. The preprocessing approach was used as an initial step in our proposed methodology. In the first step of the study, the min-max normalization approach was applied. Then, the classical AIRS system was applied on the data set. In the second step, the performance of AIRS was optimized using the improved version of the APSO algorithm (IAPSO). The K-fold cross validation was employed for training-testing of the system. The general view of the proposed approach is given in Fig. 2.

Figure 2 indicates that, we first applied the data preprocessing approach using the min_max normalization technique. Different fold cross validation techniques (*K*2, *K*3, *K*4, *K*5, and *K*10) were used on the training and testing data. The parameters of the AIRS algorithm were optimized using the IAPSO method. Figure 3 shows the application of PSO and AIRS, in combination, on the WD data set. We have used an improved version of PSO (named IAPSO) as one of the widely employed techniques for optimizing parameters in the AIRS method.

The procedure of IAPSO for AIRS describes the stage of hybridization between IAPSO and AIRS. This stage starts with the use of IAPSO by initializing the position and velocity of particles. Then, to find out the fitness value of each particle, the AIRS algorithm is executed. The position of particles that have been initialized in the IAPSO process consists of 7 values which will be used as parameters in the AIRS algorithm. The AIRS algorithm is run until the stop criteria are met, and then the accuracy value is obtained. The accuracy value is used as a fitness value of a particle and the IAPSO process is resumed until the maximum number of iterations is reached. The whole process of the proposed IAPSO method for AIRS is shown in Algorithm 1, in Appendix 2.

IAPSO architecture

This study introduces a new diagnosis model for the detection of WD. Our major hypothesis was validated using one classifier namely, AIRS, one evolutionary algorithm (PSO), one WD data set (the new one), min-max normalization approach and five different cross-validation ($K = 2$, 3, 4, 5, and 10) protocols. We briefly explained the AIRS, PSO, IAPSO, and min-max normalization approaches in the following sections. In this work, we have used 5 well-known ML techniques, namely NB, MLP, J48, RF, and H-LVQ.

Particle swarm optimization

The Particle Swarm Optimization (PSO) algorithm is a stochastic optimization strategy which cannot make any assumptions regarding the gradient of the objective function introduced by Eberhart and Kennedy [52, 53]. This method was inspired from the natural social behaviors and dynamic movements with communications of various birds, insects and fishes. Indeed, the PSO updates the position of particles using a personal best position (*pbest*) and a global best (*gbest*) [54]. More information about PSO is presented in [55–57] and Appendix 3.

The PSO as one of the optimization methods expected to perform well to improve the quality of solutions. This paper uses the application of PSO to improve the quality of solutions provided by AIRS to identify diseases accurately by adding adaptive mechanisms. In a previous study, an improved PSO was carried out by converting particle coordinates when updating the velocity, which is called Rotated Particle Swarm (RPS) [58]. The calculation process uses a matrix and hence

Table 1 The combined wart data set (CWD)

No.	Feature	Range	Type
1.	Response to treatment	Yes or No	Output (target)
2.	Gender	88 Men and 92 Women	Input
3.	Age (years)	15–67	Input
4.	Time elapsed before treatment (month)	0–12	Input
5.	The number of warts	1–19	Input
6.	Types of wart (Count)	1– Common (101), Plantar (31), Both (48)	Input
7.	Surface area of the warts ^{c} (mm^2)	4–900	Input
8.	Procedure of treatment	1 = Immunotherapy and 2 = Cryotherapy	Input

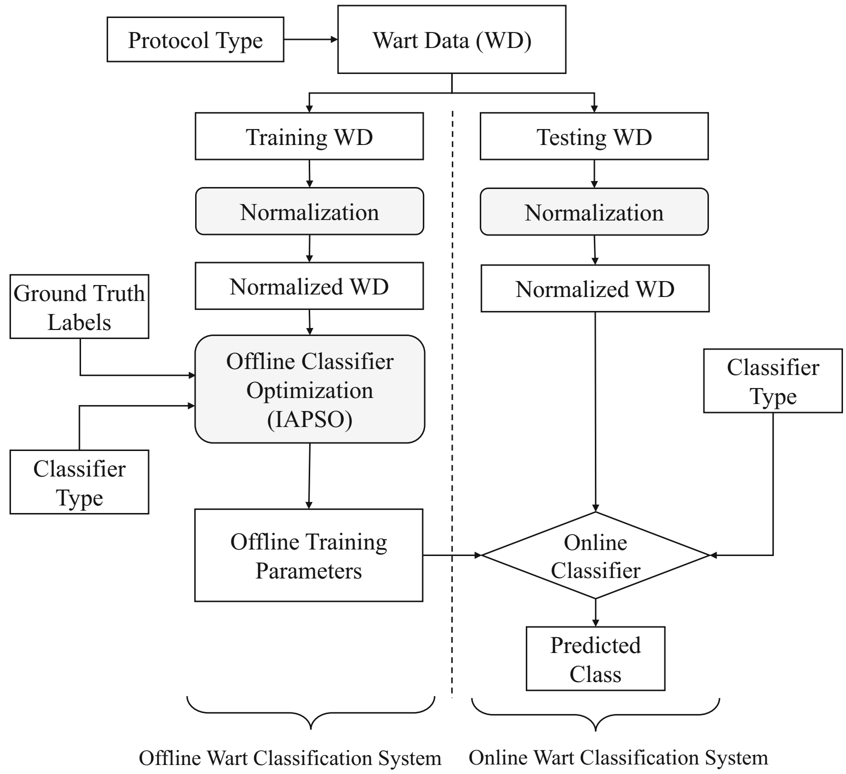


Fig. 2 The general view of the proposed system

requires a long computing time. In another study [59], a particle transfer mechanism was proposed. The transfer mechanism removes the fitness personal best value obtained previously if the average value is less than the threshold. In our opinion, this

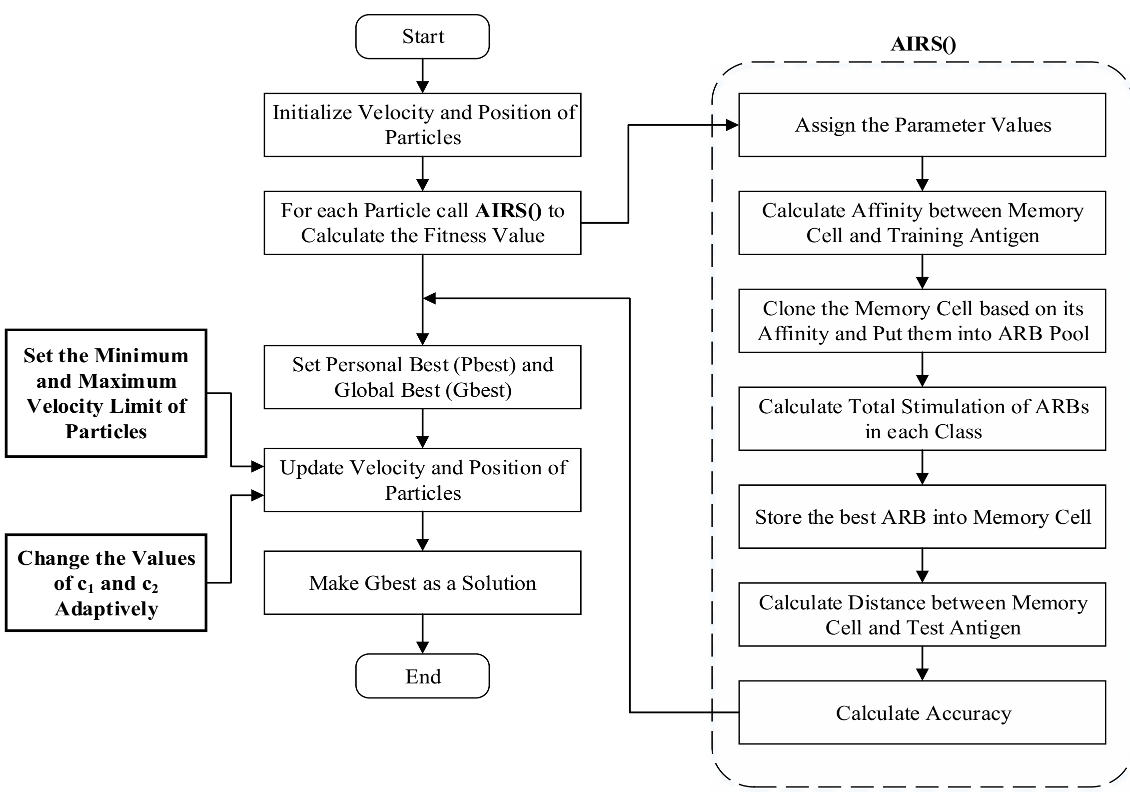


Fig. 3 Illustration of the parameter optimization procedure in the proposed IAPSO-AIRS system: the bolded boxes show how the position and velocity of each particle is updated

approach is good because its solutions can avoid local optimum solutions and use of mutation processes in genetic algorithms. However, if the particle transfer is too far away, then the particle might come out of the search area because of the selection of an incorrect new solution point. Therefore, our study uses minimum and maximum particle velocity settings to avoid particles coming out of the search area. In addition, the control parameter settings are also added to adjust particle movements adaptively according to fitness values so that particles can find better solutions in each iteration.

Improved PSO

The process of updating the velocity and position of particles is carried out using Eq. D.1and Eq. D.2. In this study, we initialized PSO by setting the minimum and maximum velocity limits for each particle in the search space. The velocity limits are influenced by the minimum and maximum values of each parameter used. Calculation of the minimum and maximum velocity limit of particles is shown in eqs. (1) and (2).

$$vmax_{\mathrm{i}} = \mathrm{k} \times \frac{(xmax_{\mathrm{i}} - x\mathrm{min}_{\mathrm{i}})}{2} \tag{1}$$

$$v\mathrm{min}_{\mathrm{i}} = -vmax_{\mathrm{i}} \tag{2}$$

where, k is a random number with a range of values [0, 1]. In this study, the value of k equal to 0.6 was used, $v\ max_i$is the maximum velocity limit of a particle in the search space, $v\ min_i$ is the minimum velocity limit of a particle in the search space, $x\ max_i$ is the maximum value limit of a parameter, and $x\ min_i$is the minimum value limit of a parameter.

The improvement of PSO was also carried out by applying random injection mechanism each iteration in multiples of 3. The mechanism was performed by generating random particles which then were inserted into the population. The purpose of the random injection mechanism is to prevent search results from being trapped at the local optimum solution.

Table 2 The summary of training for the AIRS algorithm

Parameter	Value
Threshold of Affinity	0.482
Total instances for training	180
Total replacements of memory cell	69
Mean ARB clones for each refinement iteration	94.716
Mean total resources for each refinement iteration	144.319
Mean pool size for each refinement iteration	112.225
Mean memory cell clones for each antigen	16.8
Mean ARB refinement iterations for each antigen	2.094
Mean ARB prunings for each refinement iteration	103.17

Adaptive PSO

The performance of PSO is strongly influenced by the balance of social and cognitive parameters. Determination of the parameter values c_1 and c_2 is an important factor in determining the balance parameters to produce optimal solutions. Parameters that are not well determined will affect the performance of PSO and lead to a premature convergence.

PSO as a population-based search algorithm which is often trapped in a local optimum solution. This is because there are many particles that move freely in the search space to find a solution. The poor settings of the PSO parameters used, such as particle starting points, particle velocity, and control parameters that regulate the movement of particles, causes particles to move inefficiently.

The use of adaptive mechanism in the PSO is expected to regulate particle movements during the solution search process. The control parameters in this case are the fitness values that regulate particle movements and can be changed adaptively at certain iterations by adjusting to the quality of solutions. If most of the particles in the search space have a fitness value greater than the fitness value obtained in the previous iteration, then the value of the cognitive parameters must be increased for these particles to move faster in order to get closer to the personal best of each particle. Conversely, if the particles in the search space tend to produce a fitness value that is worse than the fitness value in the previous iteration, it means that the control parameter is not able to direct the particle to a better solution so that the cognitive parameter value

Table 3 Initial results obtained using the classical AIRS algorithm applied on the new WD dataset with 180 records

Class		PRE	REC	FM	ACC (%)
*K*2	Success	0.840	**0.882**	**0.861**	–
	Failure	0.745	0.672	0.707	–
	Weighted Average	0.808	0.811	0.809	**81.11**
*K*3	Success	0.824	0.824	0.824	–
	Failure	0.656	0.656	0.656	–
	Weighted Average	0.767	0.767	0.767	76.66
*K*4	Success	**0.882**	0.815	0.847	–
	Failure	0.686	0.787	0.733	–
	Weighted Average	0.815	0.806	0.808	80.55
*K*5	Success	0.840	**0.882**	**0.861**	–
	Failure	0.745	0.672	0.707	–
	Weighted Average	0.808	0.811	0.809	**81.11**
*K*10	Success	0.825	0.874	0.849	–
	Failure	0.722	0.639	0.678	–
	Weighted Average	0.790	0.794	0.791	79.44

Bold values indicate the highest classification metrics must be decreased. In addition, if the best particle in the current iteration has a fitness value greater than the fitness value obtained in the previous iteration, then the value of the social parameter needs to be added so that the particle can move faster in order to get closer to a better solution. Conversely, if the fitness value obtained at this time is smaller than the fitness value in the previous iteration, then the particle will be directed to do a search around the point of the solution that has been obtained previously. Thus, the particles in the search space will make different adjustments to the movement so that the solution obtained is not a local optimum solution [60].

This study used an adaptive PSO mechanism to dynamically adjust the control parameters c_1 and c_2 during the search process. The adaptive PSO mechanism is based on the fitness value of each particle in each iteration, so that the values of c_1 and c_2 change adaptively. Eq. (3) is used to update c_1 and c_2 adaptively:

$$\begin{cases} c_1^{'} = c_1 + 0.05 \; and \; c_1^{''} = c_1 - 0.05 \\ c_2^{'} = c_2 + 0.05 \; and \; c_2^{''} = c_2 - 0.05 \end{cases} \tag{3}$$

where $c_1^{'}$ is used to update the value of c_1 when more than 20% of particles in the population have a new *Pbest* value that is better than the *Pbest* value in the previous iteration. If the number of particles that have a new *Pbest* value is better than the *Pbest* value in the previous iteration (should be less than 20%), then the value of c_1 will be reduced using $c_1^{''}$. The $c_2^{'}$ is used to update the value of c_2 when the *Gbest* value in the current iteration is better than the *Gbest* value in the previous iteration. Conversely, if the *Gbest* value in the current iteration does not increase, then the value of c_2 will be reduced using $c_2^{''}$.

Min-max normalization

The min-max normalization approach executes a linear transformation on the principal data [61]. This technique is a way to map a value d of P to d' in the range [new_min(p), new_max(p)]. Basically, the min-max normalization is computed using d', where *min(p)* is the minimum value of the feature and *max(p)* is the maximum value of the feature. It should be expressed that the min-max normalization approach maintains the relation among the principal data values.

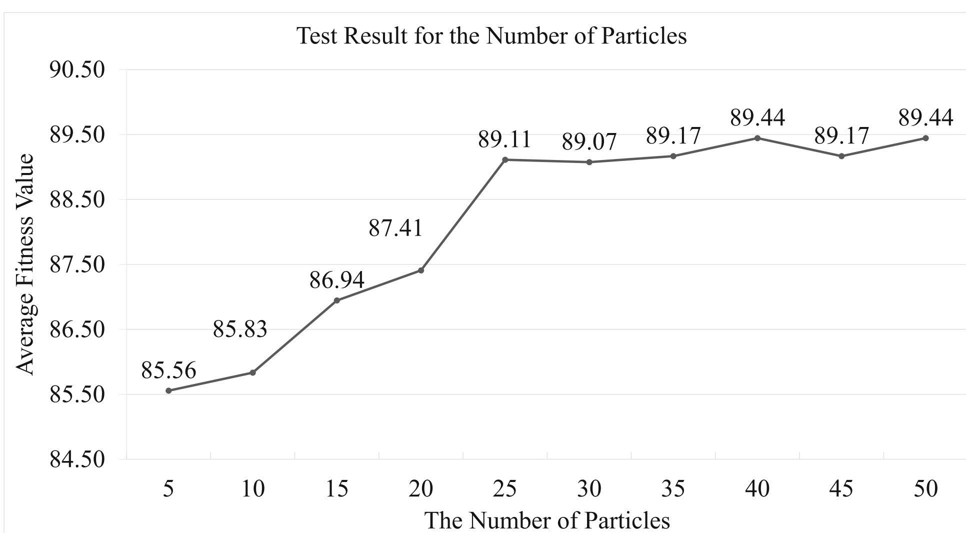
Artificial immune recognition system (AIRS)

Individual bodies have a special kind of defense system called the "immune system" [62]. This system protects our body from various diseases caused from pathogens, germs, and other toxic substances. The artificial immune system (AIS) algorithm is a computationally intelligent and rule-based machine learning method that has been widely applied to solve various complex computational or engineering problems (e. g., pattern recognition) [63–67]. The initial AIS was developed from the theoretical immunology field in mid-1980. In other words, AIS is a learning algorithm that was inspired from the human being immune system [68]. This study uses the AIRS algorithm, which is a kind of supervised learning of AIS [69]. Bai [32] showed that there is a mapping between the immune system and AIRS, which is given in [32, 62–69].

Classical classification techniques

As we discussed earlier, this work compares the performances of the proposed method with five well-known ML

Fig. 4 Test results for the number of particles



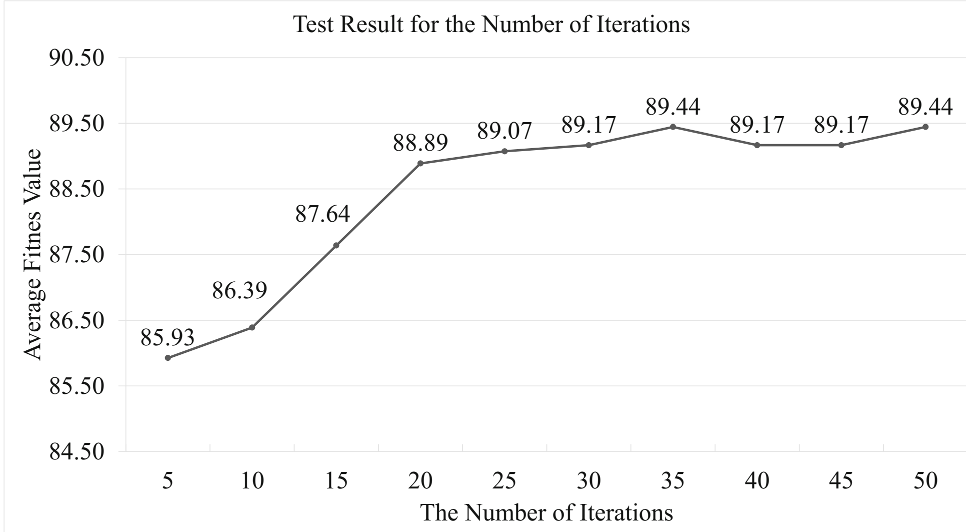


Fig. 5 Test results for the optimal number of iterations

algorithms. The algorithms used are as follows: Bayes network (BN), multilayer perceptron (MLP), J48, random forest (RF), and hierarchical LVQ (H-LVQ). More information regarding each algorithm can be found in [70–79].

Experimental protocols

In the present work, we have applied data pre-processing (min-max normalization), one classical algorithm (AIRS), and one evolutionary technique (IAPSO). We have carried out *three* experimental protocols, including (i) applying traditional AIRS on the wart data, (ii) optimization of parameters in AIRS using IAPSO, and (iii) discussion and comparison of the proposed method with other classifiers.

Experimental protocol 1: AIRS

The major objective of this section is to assess the performance of the classical AIRS method on WD and CWD data sets. In the first experiment, the AIRS algorithm was applied. The summary of training is shown in Table 2:

Due to the randomization of the data using the K-fold cross validation, the second experiment was repeated 20 times (with T = 20 trials). It should be noted that, in this step, we ran the method 20 times and then all metrics were averaged. For this step, Weka 3.8 was used as an open source data mining tool which was carried out on a PC computer equipped with Windows 7, Pentium Dual Core CPU E5300 @2.60 GHz and 2 GB RAM.

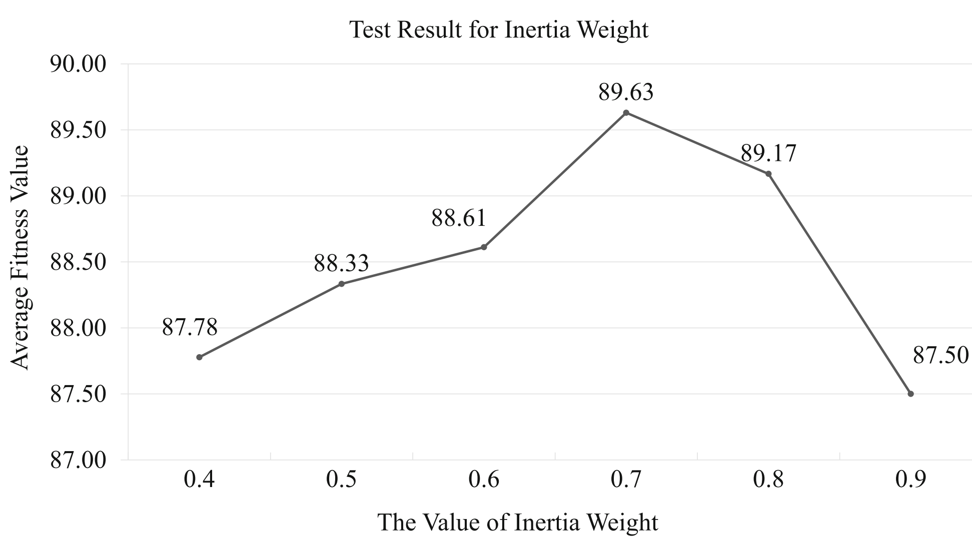


Fig. 6 Test results for the value of inertia weight

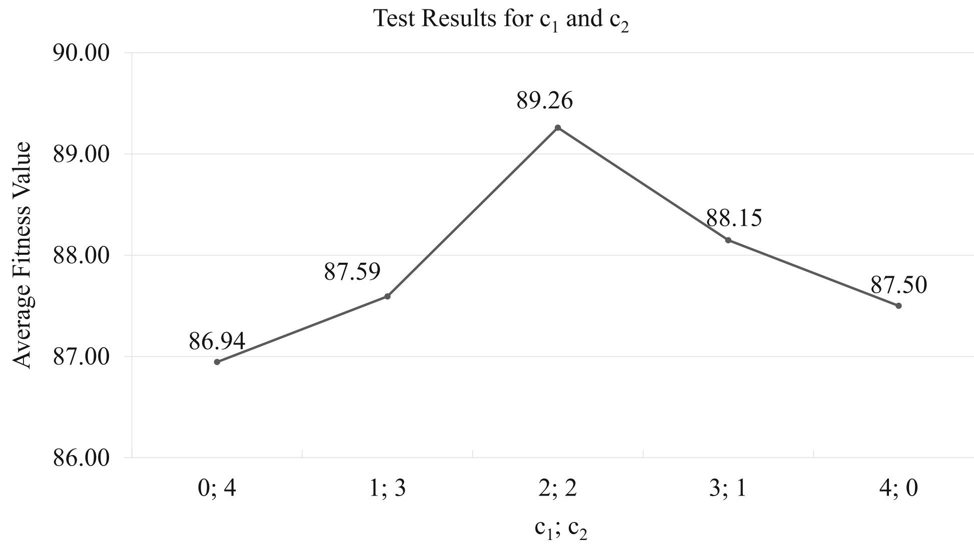


Fig. 7 Test results for the values of c_1 and c_2

Experimental protocol 2: IAPSO for AIRS

The second experiment (optimization) was implemented in Java Version: 1.8.0, when NetBeans 8.0.2 was used as IDE. Here, we used a Windows 10 operating system with Intel i5 2.4GHz CPU and 8.00 GB RAM. Several tests were performed to determine the best parameters to be used in the IAPSO for AIRS algorithm. The testing phase consisted of several testing sets and different parameter vcombinations, including the number of particles, the number of iterations, the value of inertia weight, and the values of c_1 and c_2. Testing the number of particles was used to determine the number of particles necessary to produce the optimal solution. The number of particles used in this test scenario ranged from 5 to 50. The number of iteration used was 10, the weight of inertia 0.5, and the values of c_1 and c_2 2 for both of them.

Experimental protocol 3: IAPSO for AIRS on different partition protocols

The evaluation of the performance of the proposed methodology for different partition protocols was the main objective of this section. Here, the proposed methodology was applied on the CWD data set with different partition protocols ($K2$, $K3$, $K4$, $K5$, and $K10$) and then several performance metrics such as precision (PRE), recall (REC), F-measure (FM), and accuracy (ACC), were evaluated.

Results

Results of experimental protocol 1: AIRS

In this work, we first applied the classical AISR method on the CWD and the original immunotherapy and cryotherapy data (90 records for each of the original data types), separately. It should be noted that for each partition protocol ($K=2$, 3, 4, 5, and 10), AIRS was repeated 20 times and then the averages of various measures were presented. The performance of AIRS was investigated using PRE, REC, FM, and ACC for two different classes

Table 4 Results of the classification (IAPSO for AIRS) using optimal parameters and $K10$ applied on the new data set

Trial Number	PRE	REC	FM	ACC (%)
1	0.8859	0.8762	0.8807	89.44
2	0.8826	0.8833	0.8814	88.33
3	0.8782	0.8720	0.8750	88.88
4	**0.8908**	0.8844	0.8875	**90.00**
5	0.8898	0.8735	0.8807	89.44
6	0.8839	0.8865	0.8852	89.44
7	0.8762	0.8815	0.8787	88.88
8	0.8787	0.8787	0.8787	88.88
9	0.8891	**0.8943**	**0.8916**	**90.00**
10	0.8882	0.8937	0.8908	**90.00**
11	0.8806	0.8888	0.8844	89.44
12	0.8787	0.8787	0.8787	88.88
13	0.8830	0.8858	0.8844	89.44
14	0.8752	0.8808	0.8778	88.88
15	0.8771	0.8822	0.8795	88.88
16	0.8847	0.8872	0.8859	89.44
17	0.8882	0.8937	0.8908	**90.00**
18	0.8898	0.8735	0.8807	89.44
19	0.8830	0.8858	0.8844	89.44
20	0.8806	0.8888	0.8844	89.44
Average	0.8832	0.8835	0.8831	89.33

Bold values indicate the highest values of classification metrics

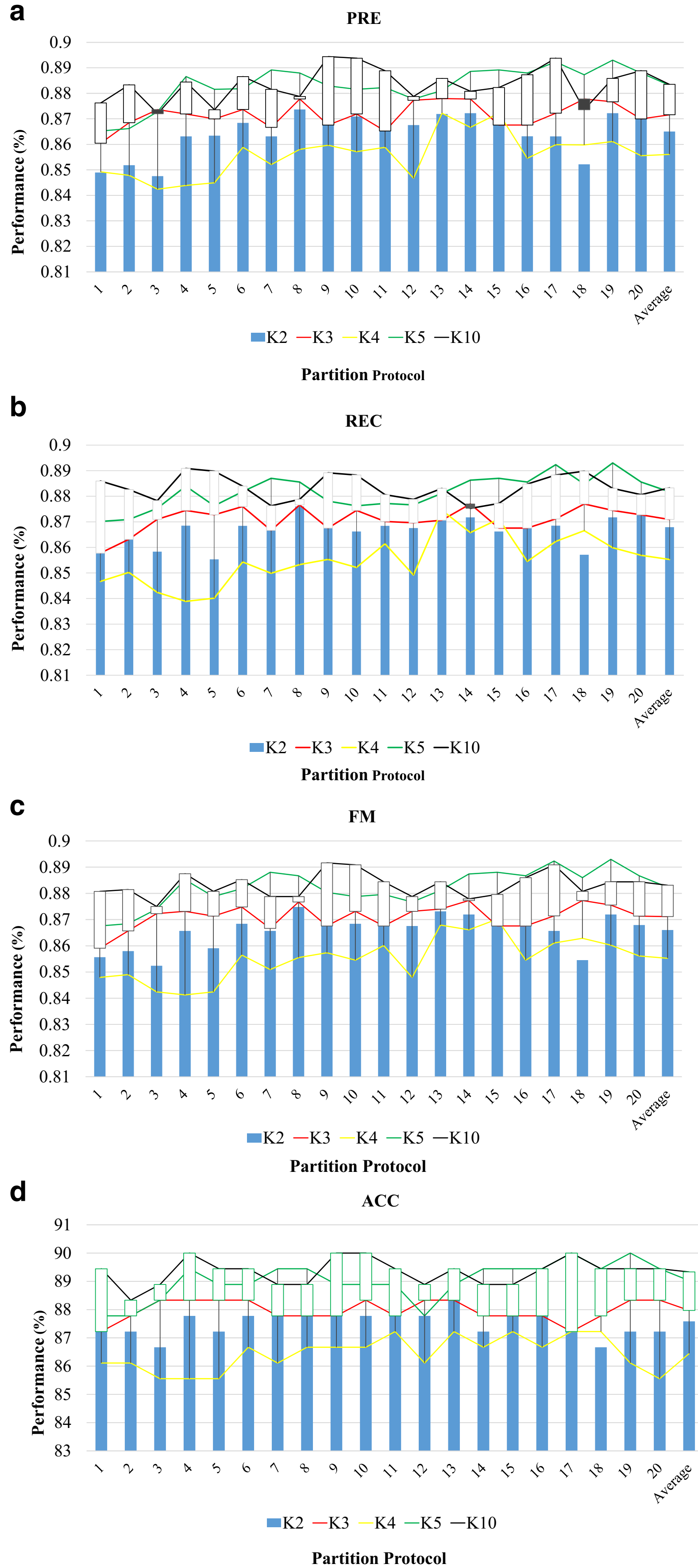


Fig. 8 Results of the classification obtained by the IAPSO method using optimal parameters and different protocols ($K = 2, 3, 4, 5,$ and 10) applied the CWD data set: **a**: PRE, **b** REC, **c** FM, and **d**: ACC. Note: Trial No. 21 is the average of all of the previous 20 trials

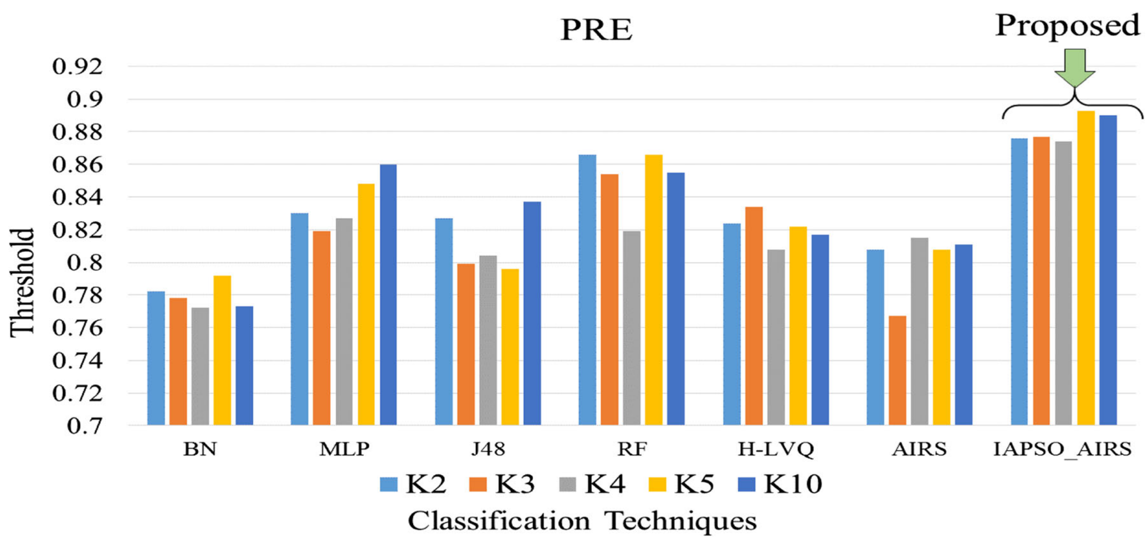


Fig. 9 Comparison of performances of different ML algorithms in terms of 'PRE' for the CWD data set

(success and failure). In the first step, AIRS was applied on separate data sets to check the performance of the algorithm (see Table 10 in Appendix 4). It can be noted from Table 10 that AIRS has yielded 91.11% (*K*4) and 82.22% (*K*2) for the cryotherapy and immunotherapy data sets, respectively. This means that the AIRS algorithm has provided comparable results according to other published works [7]. The results obtained using the classical AIRS algorithm on the CWD data set are presented in Table 3.

It can be noted from Table 3 that classical AIRS has provided a better performance for the "success" class and a weaker performance for the "failure" class. In other words, AIRS showed good classification results for only one class. Therefore, we tried to enhance the performance of AIRS using the IAPSO technique. In the rest of the study, we considered the weighted average of each protocol for the AIRS method.

Results of experimental protocol 2: IAPSO

In this experiment, the parameters of AIRS were optimized using the proposed methodology (IAPSO). To get the best values of our parameters, the proposed methodology was tested for the number of particles, the number of iterations, the value of inertia weight, and finally the values of c_1 and c_2. The test results for the number of particles are shown in Fig. 4.

It can be seen from Fig. 4 that the average fitness value increases with an increase in the number of particles. However, upon reaching a certain point, the fitness value stops augmenting. The fitness values tend to be stable when the number of particles is above 25, which means that 25 is the optimal number of particles. The second tests was for the number of iterations that necessary to produce the optimal solution. The number of iterations used in this test scenario ranged from 5 to 50. The number of particles used was 25, the

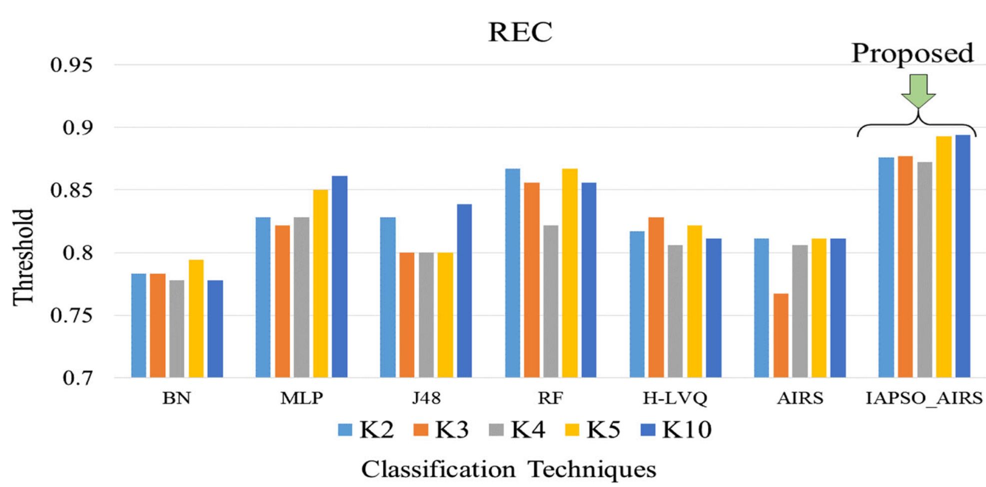


Fig. 10 Comparison of performances of different ML algorithms in terms of 'REC' for the CWD data set

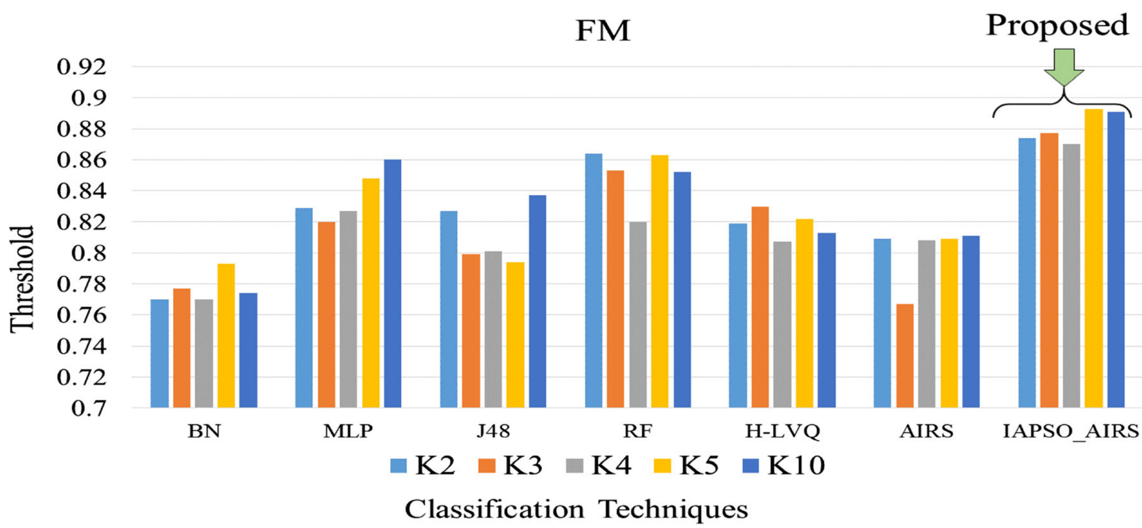


Fig. 11 Comparison of performances of different ML algorithms in terms of 'FM' for the CWD data set

weight of inertia was 0.5, and the values of c_1 and c_2 were 2. Test results for the number of iterations are shown in Fig. 5.

In Fig. 5, it can be seen that the average fitness value increases when the number of iterations increases. This increase in fitness value occurs for iterations 5 to 20 but tends to be stable when the number of iterations is above 20. This condition indicates that the number of optimal iterations is 20. In the third step, the value of inertia weight was tested. This value is used to determine the amount of inertia weights needed to produce the optimal solution. The values of inertia weights used in this test scenario were0.4, 0.5, 0.6, 0.7, 0.8, and 0.9. The number of particles used was 25, the number of iterations was 20, and the values of c_1 and c_2 were 2. Test results of this test are shown in Fig. 6.

Figure 6 shows that the best value of inertia is equal to 0.7 and the average fitness value decrease when larger values of inertia weight were applied. Larger values of inertia weight make the exploration process faster by searching globally. Hence, it gets stuck into a local optimum solution. Smaller values of inertia weights make the particles more likely to perform local searching or exploitation [72]. In this study, the best found value of inertia weight was 0.7. The last test was for the values of c_1 and c_2. The values of c_1 used in this test scenario were 0, 1, 2, 3, and 4 with the condition $c_1 + c_2 = 4$. The number of particles used was 25, the number of iterations was 20, and the value of inertia weight was 0.7. The test results for the values of c_1 and c_2 are shown in Fig. 7.

Figure 7 shows that the best parameters used to generate the optimal solution are when the values of c_1 and c_2 are 2. With a smaller value of c_1, the convergence to a certainsolution will occur quickly so that it has a tendency to produce a local optimal solution because the

Fig. 12 Comparison of performances of different ML algorithms in terms of 'ACC' for the CWD data set

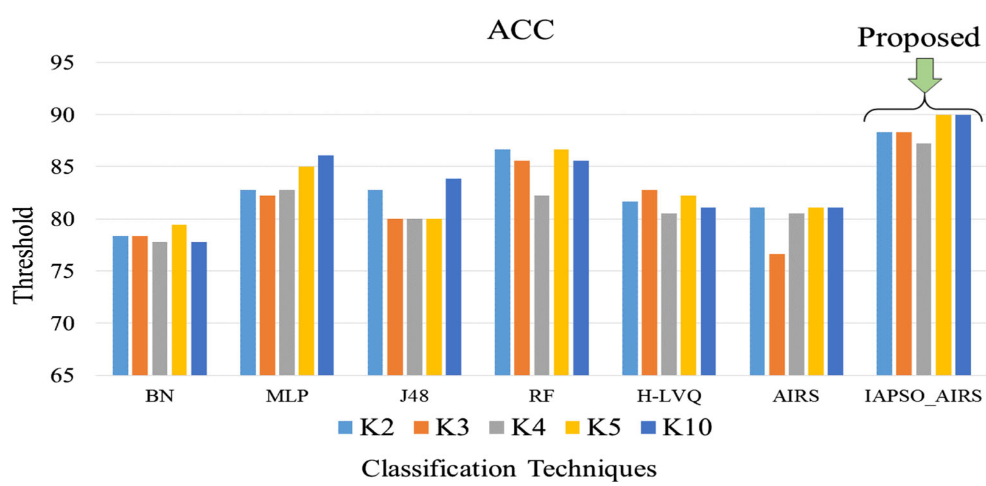


Table 5 Performance analysis of AIRS versus IAPSO for AIRS

Protocol	Improvement			
	PRE	REC	FM	ACC (%)
*K*2	0.068 ↑*	0.065 ↑	0.065 ↑	7.22 ↑
*K*3	**0.11** ↑	**0.11** ↑	**0.11** ↑	**11.67** ↑
*K*4	0.059 ↑	0.066 ↑	0.062 ↑	6.67 ↑
*K*5	0.085 ↑	0.082 ↑	0.084 ↑	8.89 ↑
*K*10	0.10 ↑	0.10 ↑	0.10 ↑	8.89 ↑
Average	0.0844 ↑	0.0846 ↑	0.0842 ↑	8.668↑

* Note: ↑ means that the value was improved after using the proposed model

Bold values indicate the highest values of classification metrics

process of updating leads to a social component. Conversely, with a smaller value of c_2, fewer particles interact with each other because the process of updating leads to a cognitive component of each particle. This causes the particles to remain in the area around their initial position [80]. Based on the test results shown in Figs. 4, 5, 6 and 7, it can be seen that the optimal parameters of IAPSO are as follows: the number of particles is 25, the number of iterations is 20, the value of the inertia weight is 0.7, the value of c_1 is 2, and the value of c_2 is also 2.

Table 4 shows an example of the classification of WD using the optimal parameters obtained from the test results. Since APSO is a stochastic algorithm, the experiment was performed 20 times to get the average values of PRE, REC, FM, and ACC. It should be noted that *K*10 helped provide the best average accuracy.

It can be noted from Table 4 that the best accuracy is 90% obtained using the proposed methodology with *K10*. Moreover, the average accuracy in Table 4 is 89.33%.

Results of experimental protocol 3: IAPSO on different partition protocols

The third experiment was conducted to evaluate the performance of the proposed methodology with different partition protocols (*K*2, *K*3, *K*4, *K*5, and *K*10). Therefore, AIPSO-AISR was applied with these *five* protocols on the CWD data. Test results for the inertia weights are shown in Fig. 8.

Fig. 8 shows that *K*10 has indicated the best average accuracy compared to other *four* protocols with an average accuracy of 89.33%. In addition, *K*10 has provided the highest average value of FM as compared to the other *four* protocols with the average of 88.3%. Moreover, we applied 5 classical ML techniques (BN, MLP, J48, RF, and H-LVQ) on the new data set to check the performance of the proposed methodology. It may be noted that for each of these classifiers, the highest value of each metric was selected. The results are illustrated in Figs. 9, 10, 11, and 12.

It can be noted from Figs. 9-12 that the proposed IAPSO for AIRS algorithm has yielded the highest PRE, REC, FM, and ACC rates as compared to classical classifiers. Thus, we could argue that the IAPSO for AIRS algorithm is very effective for an early diagnosis of WD.

Performance evaluation

Benchmarking improved APSO against AIRS

Since the improvement of AIRS was one of the main objectives of this study, we compared the performance of classical AIRS and optimized AIRS using the proposed IAPSO evolutionary method. Thus, the obtained resultsin first, second, and third experiments were compared. The improvement obtained using IAPSO with AIRS is important (see

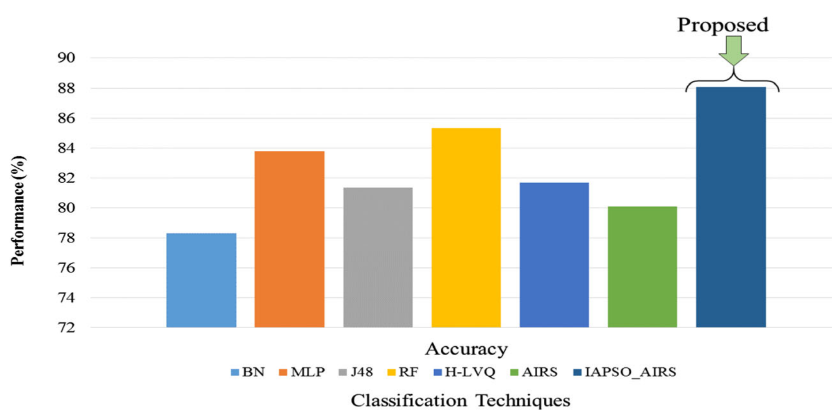


Fig. 13 Bar chart demonstrating the mean classifier accuracies for different classifiers (for K = 5, T = 20)

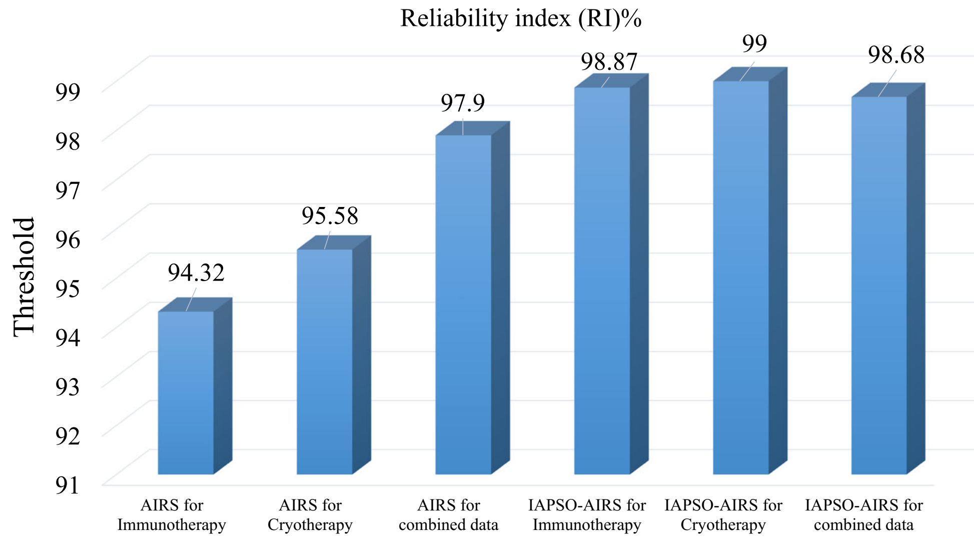


Fig. 14 Comparison of all classifiers over different data sets based on RI for $K = 2$, 3 4, 5, and 10 and $T = 20$

Table 5), which is a key point for medical and healthcare subjects.

As Table 5 shows, the accuracy of IAPSO with AIRS is much higher than that of the traditional AIRS algorithm for all protocols (*K2*, *K3*, *K4*, *K5*, and *K10*). Moreover, it can be seen that not only the accuracy values, but those of the other metrics (PRE, REC, and FM) have also improved. These results demonstrate that *K*3 achieved the maximum optimization in terms of all metrics (PRE, REC, FM, and ACC), followed by *K*10.

Moreover, the system classifier accuracy ($\eta(c)$) calculation for all parameters was evaluated by using all five sets of protocols and all trial sets for each classifier. The formula for η(c) is given by Eq. 4:

Table 6 Comparison of performances of the proposed method against previous studies using the same WD data

Author and Year	Data type / Data Size	# Selected Features (input)	Partition type	Best Classifier	ACC (%)
Akben [1] (2018)	cryotherapy / 90	6	10-fold	The fuzzy informative images + decision trees	94.40
Akben [1] (2018)	immunotherapy / 90	7	10-fold	The fuzzy informative images + decision trees	90.00
Uzun et al. [6] (2018)	cryotherapy / 90	6	NA	KNN ($K = 7$)	80.00
Uzun et al. [6] (2018)	immunotherapy / 90	7	NA	KNN ($K = 7$)	80.00
Khozeimeh et al. [7] (2017)	cryotherapy / 90	6	10-fold	Fuzzy Logic	80.70
Khozeimeh et al. [7] (2017)	immunotherapy / 90	7	10-fold	Fuzzy Logic	83.33
Guimarães et al. [81] (2018)	immunotherapy / 90	7	70% for training and 30% for testing	Fuzzy neural networks	83.33
Alizadehsani et al. [82] (2018)	cryotherapy / 90	4	NA	LIBSVM	91.11 ± 6.67
Alizadehsani et al. [82] (2018)	immunotherapy / 90	5	NA	LIBSVM	88.89 ± 6.33
Nugroho et al. [83] (2018)	Merged data / 180	6	90% for training and 10% for testing	C4.5 + RFFW	87.22
Jia et al. [84] (2019)	immunotherapy / 90	7	10-fold	LDA+ AFS + C4.5Tree	80.73 ± 0.0262
Junio Guimarães et al. [85] (2019)	cryotherapy / 90	6	10-fold	Fuzzy neural network (FNN)	88.64
Junio Guimarães et al. [85] (2019)	immunotherapy / 90	7	10-fold	Fuzzy neural network (FNN)	84.32
Proposed Method	cryotherapy / 90	6	5-fold (*K*5)	IAPSO + AIRS	94.44
Proposed Method	immunotherapy / 90	7	2-fold (*K*2)	IAPSO + AIRS	84.44
Proposed Method	CWD / 180	7	10-fold (*K*10)	IAPSO + AIRS	90.00 average (89.33)

$$\eta(c) = \frac{\sum_{k=1}^{k=K}\sum_{t=1}^{t=T}\eta(c,k,t)}{K \times T} \tag{4}$$

The results are illustrated in Fig. 13.

Reliability analysis

The reliability index (RI) of the CAD/ML system helps to evaluate the performance of a CAD/ML system. RI is the ratio of the standard deviation of the classification accuracy and the mean of the classification accuracies for all protocols. The reliability index (named ξ_N) of the system is computed using Eq. 5:

$$\xi_N(\%) = \left(1 - \frac{\sigma_N}{\mu_N}\right) \times 100 \tag{5}$$

where σ_N is the population standard deviation and μ_N is the mean of all accuracies for $K = 2$, 3 4, 5, and 10 and $T = 20$. Note that the population standard deviation can be calculated as follows:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{6}$$

where x_i represents an individual value, μ represents the mean/expected value, and N represents the total number of values. Here, we consider all accuracies obtained by different protocols for each method. For example, we carried out IAPSO with AIRS 20 times for each protocol ($K = 2$, 3 4, 5, and 10) on the new WD data set. Then, all accuracies were used to calculate RI (100 accuracies in total). The results are presented in Fig. 14.

As Fig. 14 shows, the RI values for the proposed method applied on the new WD data set is 98.68%, while the accuracy for the immunotherapy and cryotherapy data sets are 98.87% and 99%, respectively.

Discussion

Claims

This study proposed a novel evolutionary-based system to classify the WD treatment response. The proposed methodology was applied with AIRS classifier. It can be seen from Table 5 that our IAPSO method significantly improved the performance of traditional AIRS classifier in the classification of the WD treatment response.

Khozeimeh et al. [7] used the same data sets; however, they applied their methods on each data set separately. We combined *two* original data sets and created a new data set with more records, as suggested in [7], to apply various machine learning algorithms. Uzun et al. [6] investigated the same data set in their study and emphasized the need to have a bigger data set. Our model showed valuable performance results for early diagnosis of WD. The risk stratification can be improved further by adding the pathophysiological factors.

Benchmarking

Several prediction studies were carried out on the classification and diagnosis of WD. Their results are shown in Table 6.We compared our results with previous studies in the literature (see Table 6). In this regard, different aspects of each paper are considered: (i) type of data used in each paper, (ii) the size of data, (iii) the number of various features used during design of the ML system, (iv) partition type of each data, (v) type of the algorithm applied during the training/testing steps, and (vi) performance measures (metrics).

In Table 6, all reported studies have used original 90 records. Hence, we applied our new method on the same original data to compare our performance with prior studies. We have obtained the highest performance using the combination of the IAPSO and AISR methods for cryotherapy data with the accuracy of 94.44% ($K5$), while our methods yielded the second highest accuracy of 84.44% ($K2$) for immunotherapy data. In other words, our new method had the best average accuracy. Also, we have obtained the accuracy of 90% using our IAPSO-AISR method with 180 data set.

A special note on classifiers

The proposed classifier is the main part of the proposed CAD system. We have selected *six* different algorithms including AIRS, BN, MLP, J48, RF, and H-LVQ that have been tested using *five* different partitioning protocols ($K2$, $K3$, $K4$, $K5$, and $K10$). The main objective was to investigate the performance of AIRS and then compare it with other selected classifiers. To optimize the performance of AIRS, we have used an improved adaptive PSO (IAPSO) technique, added random injection, and applied the hybridization mechanism with simulated annealing. The proposed method showed the best performance compared to AIRS and other traditional classifiers. We have observed that the protocol $K10$ applied on the new data has provided the best perfomances in many cases. Our findings indicate that the IAPSO for AIRS method was the best classifier followed by RF, while the MLP and J48 classifiers have performed averagely.

Strengths, weaknesses and extensions: perspectives for future work

This research introduced the risk stratification system, called IAPSO for AIRS, which was shown to classify accurately a group of 180 WD patients affected by two types of warts: common and plantar warts. Current research presents an evolutionary-based technique that provides an average classification accuracy of 89.33% when 10-fold cross validation is used (see Table 4). Nevertheless, the proposed ML-based system still has potential for optimization. Furthermore, many other classification techniques can be used instead of AIRS. In this study only PSO was used instead of other evolutionary algorithms (EAs). Hence, in our future work, we aim to use other EAs, such as GA [86], genetic programming, covariance matrix adaptation evolution strategy (CMA-ES), and cellular evolutionary algorithm. Moreover, we aim to apply different ensemble techniques (e. g., bootstrap aggregating (bagging), boosting, stacking, voting, Bayesian model combination) either with AIRS or even with the IAPSO for AIRS algorithm. In the future, we also intend to apply different types of fuzzy logic methods, such as type-2 fuzzy logic, learning vector quantization (LVQ) technique in our proposed methodology [87–91], genetic ensembles of classifier [92] as well as different levels of genetic optimizer [93].

Conclusion

An accurate evaluation of treatment response to wart disease (WD) is a challenging task for physicians. In the current study, we investigated the treatment response in immunotherapy and cryotherapy: two popular WD treatment methods. We combined the two original wart data sets and created a new data set (each had 90 records), comprising 180 records. The proposed IAPSO method returned the precision, recall, F-measure and accuracy values of **0.8908, 0.8943, 0.8916** and **90%**, respectively, using the ***K10*** protocol. The average improvement of precision, recall, F-measure and accuracy over the artificial immune recognition system (AIRS) were **8.44%**, **8.46%**, **8.42%** and **8.668%**, respectively. In the future, the described IAPSO system can be further improved using deep learning approach if more WD-related data become available. We also intend to improve the performance of the proposed method by testing other optimization and evolutionary techniques, such as genetic algorithms, memetic algorithms, ant colony optimization, bees algorithm, artificial bee colony algorithm, and Cuckoo search using huge data.

Compliance with ethical standards

Conflict of interest Author Moloud Abdar declares that he has no conflict of interest. Author Vivi Nur Wijayaningrum declares that she has no conflict of interest. Author Sadiq Hussain declares that he has no conflict of interest. Author Roohallah Alizadehsani declares that he has no conflict of interest. Author Pawel Plawiak declares that he has no conflict of interest. Author U Rajendra Acharya declares that he has no conflict of interest. Author Vladimir Makarenkov declares that he has no conflict of interest.

Ethical approval We used two secondary datasets taken from the UCI public website (http://archive.ics.uci.edu/ml/datasets/Immunotherapy+Dataset) and (http://archive.ics.uci.edu/ml/datasets/Cryotherapy+Dataset+). No ethics approval is required for these datasets.

Animal studies This article does not contain any studies with animals performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

Appendix 1

List of Abbreviations/Symbols

Table 7 Shows the list of all Abbreviations/Symbols used in this study

SN	Abbreviations/Symbols	Description
1	ML	Machine Learning
2	DM	Data Mining
3	IDE	Integrated Development Environment
4	C	Total number of classifiers (7)
5	*K*	Total number of partition protocols (5)
6	*T*	Total number of trials (20)
7	*K2*	Protocol of partition (1/2 samples for training and 1/2 for testing)
8	*K3*	Protocol of partition (2/3 samples for training and 1/3 for testing)
9	*K4*	Protocol of partition (3/4 samples for training and 1/4 for testing)
10	*K5*	Protocol of partition (4/5 samples for training and 1/5 for testing)
11	*K10*	Protocol of partition (9/10 samples for training and 1/10 for testing)
12	AIRS	Artificial Immune Recognition System
13	IAPSO	Improved Adaptive Particle Swarm Optimization
14	ACC	Accuracy
15	PRE	Precision
16	REC	Recall
17	FM	F-Measure (F_1 Score)
18	N	Total data size for the data set
19	k-NN	k-Nearest Neighbors
20	BN	Bayes Network
21	MLP	Multilayer Perceptron
22	J48	J48 Classifier
23	RF	Random Forest
24	H-LVQ	Hierarchical LVQ
25	CAD	Computer-Aided Diagnosis
26	CDSS	Clinical Decision Support System
27	CDSA	Clinical Decision Support Algorithm
28	UCI	University of California, Irvine
29	AT	Affinity Threshold
30	RI	Reliability index
31	ξ_N	Reliability index for data set
32	σ	The population standard deviation
33	μ	The mean of all accuracies for all protocols
34	ag	The antigen
35	MC	The memory cell set
36	ClonalRate	The number of mutated clones
37	HyperClonalRate	The number of mutated clones a memory cell
38	maxstim	The maximum stimulation value
39	minstim	The minimum stimulation value
40	min(p)	The minimum value of a feature
41	max(p)	The maximum value of a feature
42	$v\ max_i$	The maximum velocity limit of a particle in the search space
43	$v\ min_i$	The minimum velocity limit of a particle in the search space
44	$x\ max_i$	The maximum value limit of a parameter
45	$x\ min_i$	The minimum value limit of a parameter
46	AFF	Affinity
47	EUC-DIS	Euclidean Distance
48	AT	A delegate of the average affinity over whole training data set
49	STI	Stimulation
50	NC	Number of Clonal
51	CR	Clonal Rate
52	HCR	Hyper Clonal Rate
53	ab.r	The constant clonal rate (ab.resource)
54	ARGM	Argmax
54	d′	The mapped value of d

Appendix 2

Flowchart of the proposed IAPSO for AIRS

```
Algorithm 1: IAPSO for AIRS
Inputs:
Wart disease data set
Outputs:
E – evaluation coefficients and confusion matrix
P – classification of wart disease (WD) treatment response

Procedure of IAPSO
For each particle
    Initialize particle position randomly
    Initialize particle velocity
End
Do
    For each particle
        Calculate fitness value using Procedure of AIRS
        If the fitness value > the personal best fitness value (Pbest)
            Set current value as the new Pbest
        End if
    End for
    Choose a particle with the best fitness value of all the particles as Gbest
    For each particle
        Update particle velocity by calculating the minimum and maximum velocity limit
        Update particle position
    End for
    If 20% Pbest in the current iteration >Pbest in the previous iteration
        Increase the value of c1
    Else
        Decrease the value of c1
    End if
    If Gbest in the current iteration >Gbest in the previous iteration
        Increase the value of c2
    Else
        Decrease the value of c2
    End if
    If iteration % 3 = 0
        Generate a new particle with random position and velocity
    End if
While maximum iteration is not attained

Procedure of AIRS
For each antigen
    Select memory cell that has highest affinity to antigen from memory cell pool
    Create a pool of B-cells, which consists of the offspring of the selected memory cell
    Do
        Clone and mutate most highly stimulated B-cell
        Remove least stimulated B-cells
    While stimulation level > threshold
    If affinity of the best B-cell > affinity of best memory cell
        Add the best B-cell to the memory pool
        Remove the memory cell from the memory pool
    End if
End for
```

Appendix 3

Initialization of particle position

In this study, a particle consists of 9 dimensions that describe the parameters to be used in the AIRS algorithm. These values are randomly selected in different value ranges for each parameter. The range of values for each parameter used in this study are presented in Table 8:

Table 8 Mapping between the immune system and AIRS

Parameter	Range
Affinity Threshold Scalar (ATS)	[0, 1]
Clonal rate	[1, 19]
Hypermutation rate	[1, 19]
Mutation rate	[0, 1]
Total resources	[150, 300]
Stimulation threshold	[0, 1]
ARB cell pool size	[0, 10]
Memory cell pool size	[0, 10]
k-NN	[1, 7]

An example of particle initialization is shown in Table 9.

Table 9 Example of a particle

ATS	Clonal Rate	Hypermutation Rate	Mutation Rate	Total Resources	Stimulation Threshold	ARB Cell Pool Size	Memory Cell Pool Size	k-NN
0.1545	10.0	18.0	0.2398	112.0	0.0709	3	5	1

Table 9 shows that a particle consists of 9 dimensions: affinity threshold scalar, clonal rate, hypermutation rate, mutation rate, total resources, stimulation threshold, ARB cell pool size, memory cell pool size, and k-NN with different values. These values are then used as parameters in AIRS.

Fitness value

The fitness function is used to find out how well the position has been found by each particle. A particle that has a high fitness value indicates that the position of the particle is close to the optimal solution. The higher the fitness value of a particle, the closer the particle position to the optimal solution. The fitness value can be obtained after applying the values of a particle to the parameters used in AIRS. AIRS will conduct a training process using these parameters. After the training process is done, the test process will be performed by calculating the suitability of the classification result with the actual class data. Therefore, the fitness value will be calculated using the accuracy, as shown in ACC. Moreover, the performances of the methods are evaluated using the other metrics such as precision, recall, and F-Measure, which are indicated as PRE, REC, and FM.

$$\begin{cases} ACC = \dfrac{TP + TN}{TP + FP + FN + TN} \& PRE = \dfrac{TP}{TP + FP} \\ REC = \dfrac{TP}{TP + FN} \& FM = \dfrac{2 \times PRE \times REC}{PRE + REC} \end{cases}$$

where, True Positive (TP) is the value obtained when the actual class of the data was 1 (Success) and the predicted was also 1 (Success), True Negative (TN) is the value obtained when the actual class of the data was 0 (Failed) and the predicted was 0 (Failed), False Positive (FP) is the value obtained when the actual class of the data was 0 (Failed) and the predicted was 1 (Success), False Negative (FN) is the value obtained when the actual class of the data was 1 (Success) and the predicted was 0 (Failed).

Appendix 4

Table 10 The initial results obtained by using the AIRS algorithm applied on the two original data sets

Class		The Cryotherapy				The Immunotherapy			
		PRE	REC	FM	ACC (%)	PRE	REC	FM	ACC (%)
K2	Success	0.745	0.905	0.817	–	**0.867**	0.915	**0.890**	–
	Failure	0.897	0.729	0.805	–	0.600	0.474	0.529	–
	Weighted Average	0.826	0.811	0.810	81.11	0.810	0.822	0.814	**82.22**
K3	Success	0.860	0.881	0.871	–	0.792	0.859	0.824	–
	Failure	0.894	0.875	0.884	–	0.231	0.158	0.187	–
	Weighted Average	0.878	0.878	0.878	87.77	0.674	0.711	0.690	71.11
K4	Success	0.870	**0.952**	0.909	–	0.827	**0.944**	0.882	–
	Failure	**0.955**	0.875	**0.913**	–	0.556	0.263	0.357	–
	Weighted Average	0.915	0.911	0.911	**91.11**	0.770	0.800	0.771	80
K5	Success	0.765	0.929	0.839	–	0.822	0.845	0.833	–
	Failure	0.923	0.750	0.828	–	0.353	0.316	0.333	–
	Weighted Average	0.849	0.833	0.833	83.33	0.723	0.733	0.728	73.33
K10	Success	0.783	0.857	0.818	–	0.790	0.901	0.842	–
	Failure	0.864	0.792	0.826	–	0.222	0.105	0.143	–
	Weighted Average	0.826	0.822	0.822	82.22	0.670	0.733	0.694	73.33

Bold values indicate the highest values of classification metrics

References

1. Akben, S. B., Predicting the success of wart treatment methods using decision tree based fuzzy informative images. Biocybernetics and Biomedical Engineering., 2018.
2. Kyriakis, K. P., Pagana, G., Michailides, C., Emmanuelides, S., Palamaras, I., and Terzoudi, S., Lifetime prevalence fluctuations of common and plane viral warts. J. Eur. Acad. Dermatol. Venereol. 21(2):260–262, 2007.
3. Hogendoorn, G. K., Bruggink, S. C., Hermans, K. E., Kouwenhoven, S. T. P., Quint, K. D., Wolterbeek, R. et al., Developing and validating the Cutaneous WARTS (CWARTS) diagnostic tool: a novel clinical assessment and classification system for cutaneous warts. Br. J. Dermatol. 178(2):527–534, 2018.
4. Karadaghy, B., Kamal, N. M., and Othman, N., Prevalence and Knowledge about Genital and other Warts among Students in Sulaimani Polytechnic University: A cross-sectional study. Kurdistan Journal of Applied Research:96–101, 2018.
5. Loo, S. K. F., and Tang, W. Y. M., Warts (non-genital). BMJ Clinical Evidence, 2009.
6. Uzun, R., İşler, Y., & Toksan, M., Choose of wart treatment method using Naive Bayes and k-nearest neighbors classifiers. In: 2018 26th Signal Processing and Communications Applications Conference (SIU). IEEE, 2018.
7. Khozeimeh, F., Alizadehsani, R., Roshanzamir, M., Khosravi, A., Layegh, P., and Nahavandi, S., An expert system for selecting wart treatment method. Comput. Biol. Med. 81:167–175, 2017.
8. Al-antari, M. A., Al-masni, M. A., Choi, M. T., Han, S. M., and Kim, T. S., A fully integrated computer-aided diagnosis system for digital X-ray mammograms via deep learning detection, segmentation, and classification. Int. J. Med. Inform. 117:44–54, 2018.
9. Saba, L., Jain, P. K., Suri, H. S., Ikeda, N., Araki, T., Singh, B. K. et al., Plaque tissue morphology-based stroke risk stratification using carotid ultrasound: a polling-based PCA learning paradigm. J. Med. Syst. 41(6):98, 2017.
10. Acharya, U. R., Faust, O., Sree, S. V., Molinari, F., Garberoglio, R., and Suri, J. S., Cost-effective and non-invasive automated benign & malignant thyroid lesion classification in 3D contrast-enhanced ultrasound using combination of wavelets and textures: a class of ThyroScan™ algorithms. Technology in Cancer Research & Treatment 10(4):371–380, 2011.
11. Boi, A., Jamthikar, A. D., Saba, L., Gupta, D., Sharma, A., Loi, B. et al., A Survey on Coronary Atherosclerotic Plaque Tissue Characterization in Intravascular Optical Coherence Tomography. Curr. Atheroscler. Rep. 20(7):33, 2018.
12. Araki, T., Ikeda, N., Shukla, D., Jain, P. K., Londhe, N. D., Shrivastava, V. K. et al., PCA-based polling strategy in machine learning framework for coronary artery disease risk assessment in intravascular ultrasound: A link between carotid and coronary grayscale plaque morphology. Comput. Methods Prog. Biomed. 128:137–158, 2016.
13. Maniruzzaman, M., Rahman, M. J., Al-MehediHasan, M., Suri, H. S., Abedin, M. M., El-Baz, A., and Suri, J. S., Accurate Diabetes Risk Stratification Using Machine Learning: Role of Missing Value and Outliers. J. Med. Syst. 42(5):92, 2018.
14. Maniruzzaman, M., Kumar, N., Abedin, M. M., Islam, M. S., Suri, H. S., El-Baz, A. S., and Suri, J. S., Comparative approaches for classification of diabetes mellitus data: Machine learning paradigm. Comput. Methods Prog. Biomed. 152:23–34, 2017.
15. Shrivastava, V. K., Londhe, N. D., Sonawane, R. S., and Suri, J. S., A novel and robust Bayesian approach for segmentation of psoriasis lesions and its risk stratification. Comput. Methods Prog. Biomed. 150:9–22, 2017.
16. Gulshan, V., Peng, L., and Coram, M., (2016). Development and Validation of a Deep Learning Algorithm for Detection of Diabetic

Retinopathy in Retinal Fundus Photographs. JAMA. 316(22): 2402–2410, 2016.
17. Nayak, D. R., Dash, R., and Majhi, B., Brain MR image classification using two-dimensional discrete wavelet transform and AdaBoost with random forests. Neurocomputing 177:188–197, 2016.
18. Majid, A., Ali, S., Iqbal, M., and Kausar, N., Prediction of human breast and colon cancers from imbalanced data using nearest neighbor and support vector machines. Comput. Methods Prog. Biomed. 113(3):792–808, 2014. https://doi.org/10.1016/j.cmpb.2014.01.001.
19. Shi, L., Xi, L., Ma, X. M., Weng, M., and Hu, X. H., A novel ensemble algorithm for biomedical classification based on ant colony optimization. Appl. Soft Comput. 11:5674–5683, 2011.
20. Arabasadi, Z., Alizadehsani, R., Roshanzamir, M., Moosaei, H., and Yarifard, A. A., Computer aided decision making for heart disease detection using hybrid neural network-genetic algorithm. Comput. Methods Prog. Biomed. 141:19–26, 2017.
21. Kim, J. K., and Kang, S., Neural network-based coronary heart disease risk prediction using feature correlation analysis. Journal of Healthcare Engineering 2017:1–13, 2017.
22. Danaee, P., Ghaeini, R., and Hendrix, D. A., A deep learning approach for cancer detection and relevant gene identification. Pac. Symp. Biocomput. 22:219–229, 2016.
23. Kourou, K., Exarchos, T. P., and Exarchos, K. P., Machine learning applications in cancer prognosis and prediction. Comput Struct Biotechnol J 13:8–17, 2015.
24. Summers, M. J., Madl, T., Vercelli, A. E., Aumayr, G., Bleier, D. M., and Ciferri, L., Deep Machine Learning Application to the Detection of Preclinical Neurodegenerative Diseases of Aging. DigitCult - Scientific Journal on Digital Cultures 2(2):9–24, 2017. https://doi.org/10.4399/97888255088952.
25. Chikh, M. A., Saidi, M., and Settouti, N., J. Med. Syst. 36:2721, 2012. https://doi.org/10.1007/s10916-011-9748-4.
26. Abdar, M., and Yen, N. Y., Design of A Universal User Model for Dynamic Crowd Preference Sensing and Decision-Making Behavior Analysis. IEEE Access 5:24842–24852, 2017.
27. Xia, Y., Liu, C., Li, Y., and Liu, N., A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring. Expert Syst. Appl. 78:225–241, 2017.
28. Hassoon, M., Kouhi, M. S., Zomorodi-Moghadam, M., & Abdar, M., Rule Optimization of Boosted C5. 0 Classification Using Genetic Algorithm for Liver disease Prediction. In: Computer and Applications (ICCA), 2017 International Conference on (pp. 299–305). IEEE, 2017.
29. Abdar, M., Zomorodi-Moghadam, M., Das, R., and Ting, I. H., Performance analysis of classification algorithms on early detection of liver disease. Expert Syst. Appl. 67:239–251, 2017.
30. Abdar, M., Yen, N. Y., and Hung, J. C. S., Improving the Diagnosis of Liver Disease Using Multilayer Perceptron Neural Network and Boosted Decision Trees. Journal of Medical and Biological Engineering:1–13, 2017.
31. Zulvia, F. E., Kuo, R. J., & Roflin, E., An Initial Screening Method for Tuberculosis Diseases Using a Multi-objective Gradient Evolution-Based Support Vector Machine and C5. 0 Decision Tree. In: Computer Software and Applications Conference (COMPSAC), 2017 IEEE 41st Annual (Vol. 2, pp. 204–209). IEEE, 2017.
32. Bai, Q., Analysis of particle swarm optimization algorithm. Computer and information science 3(1):180, 2010.
33. Zhan, Z. H., Zhang, J., Li, Y., and Chung, H. S. H., Adaptive particle swarm optimization. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 39(6):1362–1381, 2009.
34. Toğan, V., and Daloğlu, A. T., Optimization of 3d trusses with adaptive approach in genetic algorithms. Eng. Struct. 28(7):1019–1027, 2006.
35. Alkeshuosh, A. H., Moghadam, M. Z., Mansoori, I. A., and Abdar, M., Using pso algorithm for producing best rules in diagnosis of heart disease. 2017 International Conference on Computer and Applications (ICCA):306–311, 2017. https://doi.org/10.1109/COMAPP.2017.8079784.
36. Paul, A., Shill, P., Rabin, M. et al., Adaptive weighted fuzzy rule-based system for the risk level assessment of heart disease. Appl. Intell. 48:1739, 2018. https://doi.org/10.1007/s10489-017-1037-6.
37. Manikandan, R., and Saravanan, V., A novel approach on Particle Agent Swarm Optimization (PASO) in semantic mining for web page recommender system of multimedia data: a health care perspective. Multimed. Tools Appl., 2019. https://doi.org/10.1007/s11042-018-7141-8.
38. Chen, X., Yao, X., Tang, C., Sun, Y., Wang, X., and Wu, X., Detecting Parkinson's Disease Using Gait Analysis with Particle Swarm Optimization. In: Zhou, J., Salvendy, G. (Eds), Human Aspects of IT for the Aged Population. Applications in Health, Assistance, and Entertainment. ITAP 2018. Lecture Notes in Computer Science. Vol. 10927. Cham: Springer, 2018.
39. Adem, K., Hekim, M., and Demir, S., Detection of hemorrhage in retinal images using linear classifiers and iterative thresholding approaches based on firefly and particle swarm optimization algorithms. Turk. J. Electr. Eng. Comput. Sci., 2019. https://doi.org/10.3906/elk-1804-147.
40. Wang, H., and Lv, Y., Smart pathological brain detection system by predator-prey particle swarm optimization and single-hidden layer neural-network. Multimed. Tools Appl., 2016. https://doi.org/10.1007/s11042-016-4242-0 (Online).
41. Yang, C.-H., Yang, H.-S., and Chuang, L.-Y., PBMDR: A particle swarm optimization-based multifactor dimensionality reduction for the detection of multilocus interactions. J. Theor. Biol. 461:68–75, 2019. https://doi.org/10.1016/j.jtbi.2018.10.012.
42. Sakri, S. B., Rashid, N. B. A., and Zain, Z. M., Particle swarm optimization feature selection for breast cancer recurrence prediction. IEEE Access 6:29637–29647, 2018.
43. Zhang, Y., Wang, S., Sui, Y., Yang, M., Liu, B., Cheng, H., Sun, J., Jia, W., Phillips, P., and Górriz, J. M., Multivariate Approach for Alzheimer's Disease Detection Using Stationary Wavelet Entropy and Predator-Prey Particle Swarm Optimization. Journal of Alzheimer's Disease: JAD 65(3):855–869, 2018.
44. Zeng, N., Qiu, H., Wang, Z., Liu, W., Zhang, H., and Li, Y., A New Switching-Delayed-PSO-Based Optimized SVM Algorithm for Diagnosis of Alzheimer's Disease. Neurocomputing, 2018. https://doi.org/10.1016/j.neucom.2018.09.001.
45. Nekouie, A., and Moattar, M. H., Missing value imputation for breast cancer diagnosis data using tensor factorization improved by enhanced reduced adaptive particle swarm optimization. Journal of King Saud University - Computer and Information Sciences, 2018. https://doi.org/10.1016/j.jksuci.2018.01.006.
46. Wang, X., Wang, G., and Wu, Y., An Adaptive Particle Swarm Optimization for Underwater Target Tracking in Forward Looking Sonar Image Sequences. IEEE Access 6:46833–46843, 2018.
47. Sahu, C., Kumar, P. B., and Parhi, D. R., An Intelligent Path Planning Approach for Humanoid Robots Using Adaptive Particle Swarm Optimization. International Journal on Artificial Intelligence Tools 27(05):1850015, 2018.
48. Song, Z., Liu, B., & Cheng, H., Adaptive particle swarm optimization with population diversity control and its application in tandem blade optimization. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 0954406218776680, 2018.
49. Khozeimeh, F., Jabbari Azad, F., Mahboubi Oskouei, Y., Jafari, M., Tehranian, S., Alizadehsani, R., and Layegh, P., Intralesional

immunotherapy compared to cryotherapy in the treatment of warts. Int. J. Dermatol. 56(4):474–478, 2017.

50. Cryotherapy Dataset Data Set, https://archive.ics.uci.edu/ml/datasets/Cryotherapy+Dataset+. Accessed in April 2018.
51. Immunotherapy Dataset Data Set, https://archive.ics.uci.edu/ml/datasets/Immunotherapy+Dataset, Accessed in April 2018.
52. Kennedy, J., Eberhart, R. (1995). Particle swarm optimization. In Neural Networks, 1995. Proceedings, IEEE International Conference on (pp. 1942–1948). IEEE, 2011.
53. Kennedy, J., Particle swarm optimization. In Encyclopedia of machine learning (pp. 760–766). Springer US.
54. Sengupta, S., Basak, S., & Peters, R. A., Data Clustering using a Hybrid of Fuzzy C-Means and Quantum-behaved Particle Swarm Optimization. In Computing and Communication Workshop and Conference (CCWC), 2018 IEEE 8th Annual (pp. 137–142). IEEE, 2018.
55. Engelbrecht, A. P., Computational Intelligence. England: John Wiley & Sons. Inc., 2007.
56. Ahmed, H., and Glasgow, J., Swarm intelligence: concepts, models and applications. Kingston: School Of Computing, Queens University Technical Report, 2012.
57. Shi, Y., & Eberhart, R., A modified particle swarm optimizer. In Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on (pp. 69–73). IEEE, 1998.
58. Hatanaka, T., Korenaga, T., Kondo, N., & Uosaki, K., Search performance improvement for PSO in high dimensional space. In Particle Swarm Optimization. IntechOpen, 2009.
59. Kawakami, K., and Meng, Z. Q., Improvement of particle swarm optimization. PIERS Online 5(3):261–264, 2009.
60. Liu, H. R., Cui, J. C., Lu, Z. D., Liu, D. Y., and Deng, Y. J., A hierarchical simple particle swarm optimization with mean dimensional information. Appl. Soft Comput. 76:712–725, 2019.
61. Jain, Y. K., and Bhandare, S. K., Min max normalization based data perturbation method for privacy protection. International Journal of Computer & Communication Technology 2(8):45–50, 2011.
62. Meera, P. S., and Hemamalini, S., Optimal Siting of Distributed Generators in a Distribution Network using Artificial Immune System. International Journal of Electrical and Computer Engineering (IJECE) 7(2):641–649, 2017.
63. de Castro, L. N., and Timmis, J., Artificial immune systems: A novel paradigm to pattern recognition. Artificial Neural Networks in Pattern Recognition 1:67–84, 2002.
64. Draman, N. A., Wilson, C., and Ling, S., Bio-inspired audio content-based retrieval framework (B-ACRF). World Academy of Science. Eng. Technol. 29(5):785–790, 2009.
65. Timmis, J., Neal, M., and Hunt, J., An artificial immune system for data analysis. Biosystems 55(1–3):143–150, 2000.
66. Porselvi, S., Balaji, A. N., and Jawahar, N., Artificial immune system and particle swarm optimisation algorithms for an integrated production and distribution scheduling problem. International Journal of Logistics Systems and Management 30(1):31–68, 2018.
67. Thumati, B., and Sarangapani, J., Authors' Reply to "Comments on "A Novel Fault Diagnosis and Prediction Scheme Using a Nonlinear Observer With Artificial Immune System as an Online Approximator"". IEEE Trans. Control Syst. Technol. 26(1):377–378, 2018.
68. Nikoo, M. R., and Kerachian, R., Wave Height Prediction Using Artificial Immune Recognition Systems (AIRS) and Some Other Data Mining Techniques. Iranian Journal of Science and Technology, Transactions of Civil Engineering 41(3):329–344, 2017.
69. Brownlee, J., Artificial immune recognition system (airs)-a review and analysis. Center for Intelligent Systems and Complex Processes (CISCP), Faculty of Information and Communication Technologies (ICT), Swinburne University of Technology, Victoria, 2005.
70. Petitjean, F., Buntine, W., Webb, G. I., and Zaidi, N., Accurate parameter estimation for Bayesian network classifiers using hierarchical Dirichlet processes. Mach. Learn. 107(8–10):1303–1331, 2018.
71. Mahadevan, S., and Rebba, R., Validation of reliability computational models using Bayes networks. Reliab. Eng. Syst. Saf. 87(2):223–232, 2005.
72. Gardner, M. W., and Dorling, S. R., Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. Atmos. Environ. 32(14–15):2627–2636, 1998.
73. Bayat, F. M., Prezioso, M., Chakrabarti, B., Nili, H., Kataeva, I., and Strukov, D., Implementation of multilayer perceptron network with highly uniform passive memristive crossbar circuits. Nat. Commun. 9(1):2331, 2018.
74. Drazin, S., & Montag, M., Decision tree analysis using weka. Machine Learning-Project II, University of Miami, 1–3, 2012.
75. Satyanarayana, N., Ramadevi, Y., & Chari, K. K., High blood pressure prediction based on AAA using J48 classifier. In Signal Processing and Communication Engineering Systems (SPACES), 2018 Conference on (pp. 121–126). IEEE, 2018.
76. Probst, P., and Boulesteix, A. L., To Tune or Not to Tune the Number of Trees in Random Forest. J. Mach. Learn. Res. 18(181):1–18, 2018.
77. Pal, M., Random forest classifier for remote sensing classification. Int. J. Remote Sens. 26(1):217–222, 2005.
78. Waizumi, Y., Kato, N., Saruta, K., and Nemoto, Y., High speed and high accuracy rough classification for handwritten characters using hierarchical learning vector quantization. IEICE Trans. Inf. Syst. 83(6):1282–1290, 2000.
79. Janghel, R. R., Tiwari, R., & Shukla, A., Breast Cancer Diagnostic System using Hierarchical Learning Vector Quantization. In IJCA Proceedings on National Seminar on Application of Artificial Intelligence in Life Sciences, 2013.
80. Sauermann, R., Boja, D., Kirschbaum, F., and Nelles, O., Particle Swarm Optimization for Automotive Model-Based Calibration. IFAC Proceedings Volumes 43(7):733–738, 2010.
81. Guimarães, A. J., Araujo, V. J. S., de Campos Souza, P. V., Araujo, V. S., & Rezende, T. S., Using fuzzy neural networks to the prediction of improvement in expert systems for treatment of immunotherapy. In: Ibero-American Conference on Artificial Intelligence (pp. 229–240). Springer, Cham, 2018.
82. Alizadehsani, R., Abdar, M., Jalali, S. M. J., Roshanzamir, M., Khosravi, A., & Nahavandi, S., Comparing the Performance of Feature Selection Algorithms for Wart Treatment Selection. In Proceedings of The International Workshop on Future Technology (pp. 6–18), 2018.
83. Nugroho, H. W., Adji, T. B., and Setiawan, N. A., Random forest weighting based feature selection for c4. 5 algorithm on wart treatment selection method. International Journal on Advanced Science, Engineering and Information Technology 8(5):1858–1863, 2018.
84. Jia, W., Deng, Y., Xin, C., Liu, X., and Pedrycz, W., A classification algorithm with Linear Discriminant Analysis and Axiomatic Fuzzy Sets. Mathematical Foundations of Computing 2(1):73–81, 2019.
85. Junio Guimarães, A., Vitor de Campos Souza, P., Jonathan Silva Araújo, V., Silva Rezende, T., and Souza Araújo, V., Pruning Fuzzy Neural Network Applied to the Construction of Expert Systems to Aid in the Diagnosis of the Treatment of Cryotherapy and Immunotherapy. Big Data and Cognitive Computing 3(2):22, 2019.
86. Pławiak, P., and Rzecki, K., Approximation of phenol concentration using computational intelligence methods based on signals from the metal-oxide sensor array. IEEE Sensors J. 15(3):1770–1783, 2015.
87. Olivas, F., Amador-Angulo, L., Perez, J., Caraveo, C., Valdez, F., and Castillo, O., Comparative study of type-2 fuzzy particle swarm, bee colony and bat algorithms in optimization of fuzzy controllers. Algorithms 10(3):101, 2017.

88. Valdez, F., Vazquez, J. C., Melin, P., and Castillo, O., Comparative study of the use of fuzzy logic in improving particle swarm optimization variants for mathematical functions using co-evolution. Appl. Soft Comput. 52:1070–1083, 2017.
89. Gaxiola, F., Melin, P., Valdez, F., Castro, J. R., and Castillo, O., Optimization of type-2 fuzzy weights in backpropagation learning for neural networks using GAs and PSO. Appl. Soft Comput. 38: 860–871, 2016.
90. Olivas, F., Valdez, F., Castillo, O., and Melin, P., Dynamic parameter adaptation in particle swarm optimization using interval type-2 fuzzy logic. Soft. Comput. 20(3):1057–1070, 2016.
91. Melin, P., Amezcua, J., Valdez, F., and Castillo, O., A new neural network model based on the LVQ algorithm for multi-class classification of arrhythmias. Inf. Sci. 279:483–497, 2014.
92. Pławiak, P., Novel genetic ensembles of classifiers applied to myocardium dysfunction recognition based on ECG signals. Swarm and Evolutionary Computation 39:192–208, 2018.
93. Książek, W., Abdar, M., Acharya, U. R., and Pławiak, P., A novel machine learning approach for early detection of hepatocellular carcinoma patients. Cogn. Syst. Res. 54:116–127, 2019.