Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Knowledge-Based Systems 235 (2022) 107629

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Coronavirus herd immunity optimizer with greedy crossover for feature selection in medical diagnosis

Mohammed Alweshah [a,*], Saleh Alkhalaileh [d], Mohammed Azmi Al-Betar [b,c], Azuraliza Abu Bakar [d]

[a] Prince Abdullah Bin Ghazi Faculty of Information and Communication Technology, Al-Balqa Applied University, Al-Salt, Jordan
[b] Artificial Intelligence Research Center (AIRC), College of Engineering and Information Technology, Ajman University, Ajman, United Arab Emirates
[c] Department of Information Technology, Al-Huson University College, Al-Balqa Applied University, Al-Huson, Irbid, Jordan
[d] Center for Artificial Intelligence Technology, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi, Selangor, Malaysia

ARTICLE INFO

Article history:
Received 9 February 2021
Received in revised form 13 August 2021
Accepted 17 October 2021
Available online 29 October 2021

Keywords:
Medical diagnosis
Feature selection
Greedy crossover
Optimization
Coronavirus herd immunity optimizer

ABSTRACT

The importance of medical data and the crucial nature of the decisions that are based on such data, as well as the large increase in its volume, has encouraged researchers to develop feature selection (FS)-based approaches to identify the most relevant data for specific medical problems In this paper, two intelligent wrapper FS approaches based on a new metaheuristic algorithm named the coronavirus herd immunity optimizer (CHIO) were applied with and without the incorporation of a greedy crossover (GC) operator strategy to enhance exploration of the search space by CHIO. The two proposed approaches, CHIO and CHIO-GC, were evaluated using 23 medical benchmark datasets and a real-world COVID-19 dataset. The experimental results indicated that CHIO-GC outperformed CHIO in terms of search capability, as reflected in classification accuracy, selection size, F-measure, standard deviation and convergence speed. The GC operator was able to enhance the balance between exploration and exploitation of the CHIO in the search and correct suboptimal solutions for faster convergence. The proposed CHIO-GC was also compared with two previous wrapper FS approaches, namely, binary moth flame optimization with Lévy flight (LBMFO_V3) and the hyper learning binary dragonfly algorithm (HLBDA), as well as four filter methods namely, Chi-square, Relief, correlation-based feature selection and information gain. CHIO-GC surpassed LBMFO_V3 and the four filter methods with an accuracy rate of 0.79 on 23 medical benchmark datasets. CHIO-GC also surpassed HLBDA with an accuracy rate of 0.93 when applied to the COVID-19 dataset. These encouraging results were obtained by striking a sufficient balance between the two search phases of CHIO-GC during the hunt for correct solutions, which also increased the convergence rate. This was accomplished by integrating a greedy crossover technique into the CHIO algorithm to remedy the inferior solutions found during premature convergence and while locked into a local optimum search space.

1. Introduction

It is hard for a human to retrieve essential information from the large volume of data stored and disseminated by numerous health research centers around the world, which adversely affects the capacity of medical staff to extract the necessary knowledge from medical data [1]. Moreover, given that the health and medical care industry is one of the key industries that does not tolerate incorrect actions being taken as a result of inaccurate data processing, many artificial intelligence approaches have been used to improve the consistency of medical information as these approaches are capable of handling the growing amount of data in a highly efficient manner [2,3]. Through the use of artificial intelligence, including machine learning, it is envisaged that it will be possible to create healthcare applications that can perform as well or better than human physicians in certain tasks [4,5]. Machine learning is able to link, analyze and present data in a more intelligible manner, which then enables human medical practitioners to make accurate decisions and take appropriate action [6,7].

Information extracted by artificial intelligence techniques involves the use of a conceptual relationship that expresses the data in a new manner that is more understandable and meaningful to the data owner, without making any assumptions about what the knowledge within that data could be [8,9]. However, physical examination is a crucial task, and failure for it can affect the safety and efficiency of the overall treatment process [10,11]. There is a

* Corresponding author.
E-mail address: weshah@bau.edu.jo (M. Alweshah).

https://doi.org/10.1016/j.knosys.2021.107629
0950-7051/

lot of data that a classifier must take into consideration, although it is not related to the problem of the study and is not related to it [12,13]. Therefore, the selection of appropriate data or features to which to apply the classifier will increase the efficiency of the results produced by the classifier and at the same time reduce the time consumed by the learning model, especially when the volume of data is large [14,15]. Thus feature selection (FS) is considered a critical process for enhancing the efficiency of a learning algorithm [16].

The FS problem essentially involves finding a way to select the lowest number of relevant features from the original dataset that often comprises a massive variety of features [17]. In a large dataset; certain features are linked to the problem of interest while others are not. If all the features were chosen, this would definitely have an effect on the search results either in terms of time consumed or classification accuracy [18]. Thus, the objective of the FS process is to reduce the dimension of the search space as much as possible, but not at the cost of accuracy [19]. Therefore, the success of the selection task relies on two essential aspects: decreasing the number of features and increasing classification accuracy [20,21].

The FS process involves a generation process, evaluation phase, meeting the termination criterion and completing a validation procedure [22]. These four stages can be achieved by applying a FS method, such as a filter, wrapper, embedded or hybrid method. The wrapper method is distinct from the filter method in that it employs a learning algorithm during the evaluation phase, whereas the filter method evaluates specific features independently of the classification process by using a certain standard threshold [23,24]. The embedded method is similar to the wrapper method as a classifier is used in the selection process in the evaluation step, but the use of the classifier in the embedded method is comparatively less cost-effective than in the wrapper method [25]. On the other hand, the hybrid method sequentially employs a filter and a wrapper method and hence the selection of features involves two iterations. First, the filter is used to produce a subset of features and then the wrapper is used to pick features from a subset obtained from first step [26].

During the generation phase, a set of features is selected from the full dataset to decide whether it matches the solution or not [27]. Basically, each feature is examined to create the best subset, either through a process of forward selection or backward elimination, which increases the degree of complexity by 2^n [28]. To reduce the time it takes to generate feature subsets in this phase, an optimization method is often used as a search strategy. Optimization techniques are estimation processes and the results obtained by these techniques are either optimal or suboptimal. One of the well-known and widely used optimization methods is the metaheuristic algorithm [29–32].

Metaheuristic algorithms are intelligent algorithms that are based on the concept of identifying a particular mathematical tool with the aim of optimizing a specific problem [33]. Improvements are made by several frequent implementation attempts in order to find the correct solution for a particular problem [34]. These intelligent algorithms utilize the knowledge gathered during the search to guide the search process, during which they iteratively create new solutions by integrating one or more good solutions, and they are often also combined with some kind of operator in order to prevent them becoming stuck in a local optimum [35]. While metaheuristics try to find the optimal solution, they are typically imperfect mechanisms as they do cannot ensure that the best global solution is found. Rather, they often produce approximate results [36].

Two types of search are performed by metaheuristic algorithms to find an optimal solution: exploration and exploitation [37]. In the exploration phase, the search traverses numerous sites and different environments to explore and discover more areas for high-quality solution. Population-based metaheuristic algorithms are exploration-oriented [38,39]. On the other hand, in the exploitation phase, existing resources are focused on a particular search area. Single-based metaheuristic algorithms are considered to be exploitation-oriented [40–42].

Numerous metaheuristics have been proposed and are widely used to solve FS problems in different research domains. These include the monarch butterfly optimization algorithm (MBO) [43–45], chaotic dragonfly algorithm (CDA) [46], whale optimization algorithm (WOA) [47], spotted hyena optimizer (SHO) [48], atom search optimization (ASO) [49], chaotic interior search algorithm, equilibrium optimizer algorithm (EOA) [50], and chaotic competitive swarm optimization (CCSO) [51] among many others [52–59].

In this study, a new metaheuristic algorithm named the coronavirus herd immunity optimizer (CHIO), which was developed by Al-Betar et al. [60] in 2020, is implemented to solve FS problems in the medical diagnosis domain. The CHIO simulates herd immunity, which is considered to be a means to combat a viral pandemic. It was inspired by the coronavirus known as SARS-CoV-2 or COVID-19 which caused a global pandemic during 2020. The extent of the spread of coronavirus infection depends on how infected individuals communicate directly with other community members and herd immunity can prevent other people from acquiring the infection. In the current study, the CHIO is implemented in two different ways to select the most effective features in medical datasets. First, it is applied in its original form. Then, the exploration capability of the CHIO is enhanced by using a greedy crossover (GC) operator in an approach named CHIO-GC. The two proposed approaches are applied in a wrapper model using a K nearest neighbor (KNN) classifier, and evaluated using 23 medical benchmark datasets, as well as a COVID-19 dataset as a case of a real-world problem dataset. In addition, the performance of the two proposed approaches is compared against other methods in the literature. The analysis revealed that integrating a greedy crossover technique into the CHIO algorithm produced results that were more accurate than those produced using CHIO in its original form. This indicates that CHIO-GC has the ability to remedy the inferior solutions found during premature convergence and while locked into a local optimum search space.

The remainder of this paper is structured as follows: the works most important to this study are presented in Section 2. The suggested FS methods, CHIO, and CHIO-GC, are discussed in Section 3, 4, and 5, respectively. The tests and the findings are discussed in Section 6. Lastly, the conclusion and some possible directions for study are discussed in Section 7.

2. Related work

Feature selection has been used in a wide range of problems, including image processing, sentiment analysis, intrusion detection, and language identification as well as many other domains [61–67]. However, one of the challenges that still needs be overcome in respect of the use of FS process is its use in the field of medical diagnosis, which is the focus of this research. Therefore, in this section, the most recent work on the use of FS in medical diagnosis will be reviewed. Several different approaches have been proposed in this regard.

For instance, Li et al. [68] employed a hybrid approach for FS in medical diagnoses using a genetic algorithm (GA) to produce sustainable initial positions and a gray wolf optimization (GWO) to modify the existing population positions in a discrete search area. In experiments, the proposed approach was applied to disease diagnosis problems, and the results demonstrated that the

suggested hybrid approach is superior in terms of classification accuracy as compared to the original GA and the GWO. On the other hand, Zuo et al. [69] suggested using a filter-based FS method that explicitly uses the Menger curvature to rate all the features in an electronic health records dataset. The results showed that after reducing the number of features, high classification accuracy is achieved by this method as compared to previous methods.

Anter and Ali [70] designed a hybrid FS solution that combines the crow search optimization algorithm with chaos theory and fuzzy c-means (CFCSA). The suggested CFCSA uses the global optimization approach to prevent local minima trapping and chaos theory to resolve the lack of CSA convergence. Experiments showed that the CFCSA outperforms in terms of mean fitness and standard deviation as compared to other methods such as bat algorithm and the binary crow search algorithm.

In another work, Wang and Chen [71] developed a hybrid learning system that employs a WOA that blends chaotic and multi-swarm techniques to concurrently tackle parameter optimization and FS, as well as to optimize the efficiency of a support vector machine (SVM) to diagnose various diseases. However, the results indicated that the SVM generated by the proposed approach is actually inferior to other profitable SVM methods based on the original WOA, particle swarm optimization (PSO), bacterial foraging optimization and the GA in terms of both classification accuracy and selection size.

Furthermore, Rostami et al. [72] proposed a FS model that incorporates the idea of using node centrality and the PSO algorithm. The proposed scheme consists of three main processes. In the first step, the initial features are visualized as a graphic representation model. In the next step, the core features of all the nodes in the graph are determined. Finally, the enhanced PSO-based search method is used to pick the final features Experiments were done on five medical datasets and the results showed that the proposed approach improves on the reliability and efficacy of previous related approaches.

In research conducted by Verma et al. [73], a cost-sensitive medical diagnostic is regarded as a FS problem, in which each test provides a feature that to be used in a prediction model. The aim of their study was to identify FS methods that have the optimal balance between accuracy and cost. To this end, the researchers used the "weak dominance" problem property to create online algorithms that define a collection of features in order to offer an "optimal" trade-off between the cost and accuracy of prediction without including knowledge of the true features of the medical state. The findings confirmed the efficiency of the proposed method in respect of optimization problems generated by real-world datasets. Moreover, the FS process was also applied in [74] on a skin disease dataset by different classification techniques. The FS process in this approach enhances the dermatological prediction accuracy.

Kuppuchamy and Mangayarkarasi [75] concentrated on using fuzzy entropy to assess the importance of the feature in the diagnosis of breast cancer. In their study, a number of FS strategies were implemented to obtain useful subsets of features. In addition, the radial base function network was used as a classifier. The Wisconsin Breast Cancer dataset was used in the experiment and the findings showed that high classification accuracy was achieved with reduced selection size. The Wisconsin Diagnostic Breast Cancer dataset was also utilized by Rahman and Muniyandi [76] in their work on selecting effective features by using a FS technique. They used a 15-neuron neural network to classify the cancer. The results showed a significant improvement of up to 99.4% in classification accuracy in comparison with other methods. Another FS technique was proposed by de Lima et al. [77]. The researchers' technique was based on a twin-bound support vector machine (FSTBSVM). The experiment revealed that the proposed method is very effective and capable of delivering good results with limited features as compared to using the original datasets.

A metaheuristic algorithm was applied for FS in relation to medical issues by Too and Mirjalili [78]. Specifically, the researchers implemented a hyper-learning binary dragonfly algorithm (HLBDA) in a wrapper FS approach to find optimum feature subsets from over 21 datasets as well as a COVID-19 dataset. The findings revealed the supremacy of HLBDA in terms of increasing the classification accuracy and reducing the number of features chosen in comparison with eight previous works.

On the other hand, Abu Khurmaa et al. [79] improved the moth flame optimization (MFO) algorithm in two directions. In the first, eight binary variants are generated using eight transition functions. In the second, a Lévy flight operator is incorporated into the MFO structure in association with the transition functions, and named LBMFO-V3. It was shown that the suggested LBMFO V3 method is able to greatly outpace several well-known wrapper methods in 83% of datasets. Also, the suggested methodology surpasses other approaches in the literature approaches in 75% of the datasets. Also, a comparison with the filter-based methodology indicated that the proposed LBMFO-V3 approach is superior across 70% of the datasets.

From the above overview of related works, it can be seen that many metaheuristic algorithms have been used to solve FS problems in the medical diagnosis domain. The findings of previous research studies have shown that these smart algorithms can identify the best related features that can maximize classification accuracy. The effectiveness of these algorithms is attributed to the consistency of their random search mechanism and to their ability to strike a balance between local and global search processes.

In light of the above, in this study, two intelligent FS wrapper strategies based on a new metaheuristic algorithm called CHIO were applied with a greedy crossover operator strategy to enhance CHIO exploration for FS in the field of medical diagnosis. A crossover strategy permits individuals to exchange genetic information during the development of subsequent generations of individuals. A greedy algorithm is a step-by-step method that guarantees that the following step delivers the greatest possible value on the way to a solution. It has been shown that optimization problems can also be solved by using a greedy method [80]. This is because a greedy algorithm can eventually remove the problem if better judgments can be made at any phase, and then an optimal solution to the entire problem can be discovered.

3. Coronavirus herd immunity optimizer (CHIO)

The CHIO is a new metaheuristic algorithm that was proposed by Al-Betar et al. in 2020 [60]. Similar to many other metaheuristic algorithms, it mimics the behavior of a natural entity, in this case taking its inspiration from a pathogenic coronavirus. The CHIO imitates the process of achieving natural immunity in a herd through the implementation of herd psychology, which is known to be one of the methods of obtaining immunity from infectious diseases.

In 2020, for the third time in as many decades, a pathogenic coronavirus crossed species to infect the human population. The virus, unofficially labeled 2019-nCoV, was first observed in Wuhan, China, in people who had been exposed to seafood or a wet market [81]. The swift response of the Chinese public health and scientific community contributed to the recognition of the related clinical disease and provided initial awareness of the epidemiology of the infection [81]. Acquired immunity is developed by a human getting a normal infection via a pathogen

or by receiving an injection, often a vaccine. Herd immunity stems from the effect of the extent of human immunity on the larger herd [82]. It can be described as indirect immunity against infection that is given to susceptible persons when a reasonably large proportion of individuals in the population are resistant to the infection [62].

Herd immunity relies upon the period a disease remains inside an infected host and the pace at which the disease spreads. The introduction of a single infected person into a group of already vulnerable individuals would result in the continuous indiscriminate spreading of a disease among any of those who were approached by the infected person before such infected individuals died or recovered. The estimated number of individuals who become infected in such a vulnerable population is the so-called simple reproduction number [83]. The disease would be spread to other susceptible contacts by the persons who had acquired the disease from the original infected person, and this mechanism would repeat itself until the disease infected the whole population [84]. However, the presence of herd immunity could lead to the complete elimination of the disease from a society, and, as long as any member of the population has immunity to the disease, the potential of the disease to spread would decline. The decline in the rate of disease spread would be dependent on the size of the immunized herd. Nevertheless, even if total herd immunity could not be achieved, the effects of the disease could be mitigated by the presence of a "buffer" of resistant individuals [85].

The idea of coronavirus herd immunity was mathematically modeled by Al-Betar et al. to create a theoretical optimization algorithm named the CHIO. The model is based on the finding a way to best protect humanity against disease by converting the bulk of the helpless non-infected population into a robust population [60]. As a consequence, all the remaining vulnerable cases would not be affected and the resistant population would no longer transmit the disease. In the model, the population of herd immunity individuals are classified into three categories: susceptible, contaminated (or confirmed) and immunized (or recovered) persons [60,86]. A susceptible individual is a person who is not born with or afflicted with the virus. However, a vulnerable individual may be contaminated by interaction with infectious individuals who have refused to comply with the recommended social distancing or gap. An infected individual is a person who can spread the virus to susceptible individuals who are in close contact with the psychological distancing factor. An immunized individual is a person who is protected from infection and does not threaten untreated individuals. Therefore, this type of individual can help prevent the spreading of the virus to others and thereby avert the triggering of a pandemic [64].

Fig. 1 provides an illustration of the population hierarchy in the herd immunity scenario and the effect on acquiring immunity on the above-described three categories of individuals in the population.

It can be seen from Fig. 1 that herd immunity can be depicted as a tree in which the infectious individual is at the base or root and the branches correspond to the other individuals contacted. The right-hand portion of Fig. 1 shows that the virus cannot be spread to contacted persons if the root individual is immunized.

The herd immunity strategy can be modeled as an optimization algorithm that consists of six main phases [50]. Each of these phases is discussed in turn below:

Phase 1: Initialization

The CHIO parameters and the issue of optimization are addressed in this step. In respect of objective functionality, the optimization problem is formulated as shown in Eq. (1):

$$\text{Min } f(x) \qquad x \in \{Lb, Ub\} \tag{1}$$

where $f(x)$ is the measured objective function (or immunity rate) that is computed for the individual $x_i = (x_1, x_2, \ldots, x_n)$, where x_i is the gene indexed by i, and n represents the number of genes in each individual. Notice that each gene's value range is $xi \in [lbi, ubi]$, where lbi is located. The highest and lowest boundaries of gene xi are expressed by Lbi and Ubi. The CHIO algorithm has four algorithmic parameters and two operational parameters. The four algorithmic parameters are (1) C_0, which is the number of preliminary cases of infection initiated by one individual; (2) HIS, which is the size of the population; (3) Max_Itr, which is the actual number of iterations; and (4) n, which represents the problem dimensionality.

In this stage, two major control parameters of the CHIO are initialized: (1) the basic reproduction rate (BRr), which regulates the operators of the CHIO by propagating the coronavirus among the individuals, and (2) the maximum age of infected cases (Max_Age), which determines the classification of the infected cases as either having recovered or died.

Phase 2: Generate initial herd immunity population

The CHIO produces a set of cases (individuals) that is equal to HIS spontaneously (or heuristically). In the herd immunity population (HIP), the generated cases are stored as a two-dimensional matrix of size $n \times HIS$ as follows:

$$HIP = \begin{bmatrix} x_1^1 & x_2^1 & x_n^1 \\ x_1^2 & \overset{2}{2} & x_n^2 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \overset{HIS}{1} & x_2^{HIS} & x_N^{HIS} \end{bmatrix} \tag{2}$$

in which each row j represents a case $x\ j$ that is generated basically. This includes $x_i^j = Lbi + (Ubi - Lbi) \times U(0, 1), \forall i = 1, 2, . , . n$. The objective function (or immunity rate) is determined by using Eq. (1) for each situation. In addition, the HIS duration status variable (S) for all HIP cases is initiated by either zero (susceptible case) or one case (infected case). Note that the random initiation of the number of ones in (S) is as many as C_0.

Phase 3: Evolve coronavirus herd immunity

The evolution phase is the CHIO's primary enhancement loop, where gene x_i^j in case x^j, according to the proportion of the BRr, either remains the same or changes according to the influence of social distancing based on the following three rules for infected, susceptible and immune cases:

$$\xrightarrow{x_i^j(t+1)} \begin{cases} x_i^j(t) & r \geq BRr \\ C\left(x_i^j(t)\right) & r < \frac{1}{3} \times BRr \quad \text{(infected)} \\ N\left(x_i^j(t)\right) & r < \frac{2}{3} \times BRr \quad \text{(susceptible)} \\ R\left(x_i^j(t)\right) & r < BRr \quad \text{(immune)} \end{cases} \tag{3}$$

where r produces a number generator between 0 and 1. The three rules are described below:

1. Infected case

Under the spectrum of $r \in [0, \frac{1}{3}BRr]$ any social gap is caused by the new gene value of $x_i^j(t+1)$, which is determined by the discrepancy between the present gene and a gene obtained from a contaminated case x^c, such as

$$x_i^j(t+1) = C(x_i^j(t)) \tag{4}$$

where

$$C\left(x_i^j(t)\right) = x_i^j(t) + r \times (x_i^j(t) - x_i^c(t)) \tag{5}$$

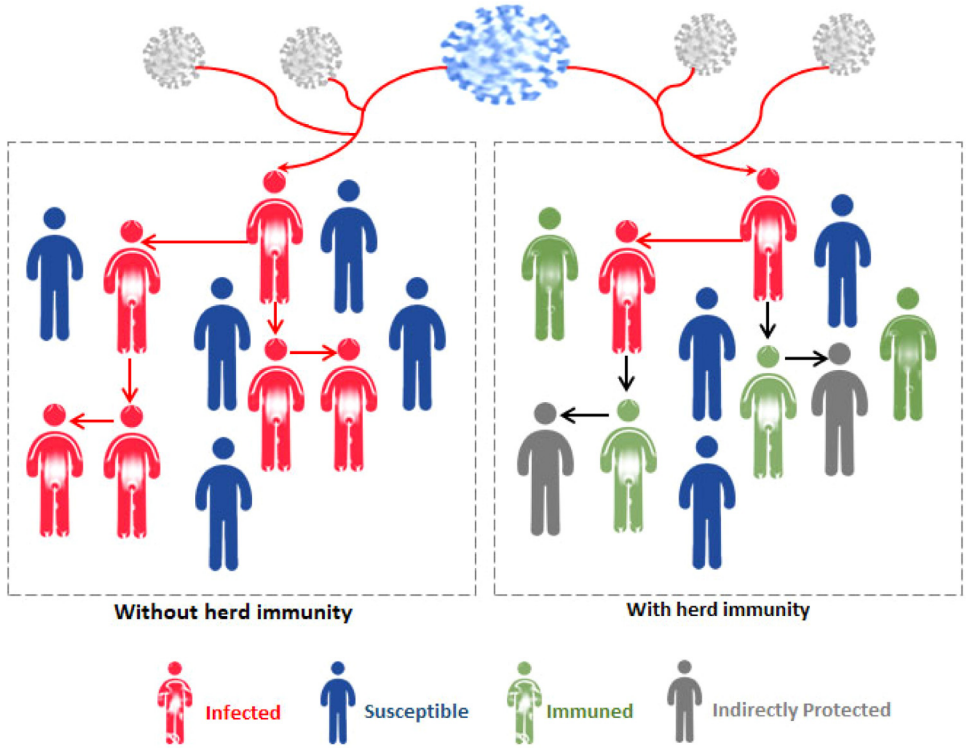

Fig. 1. Population hierarchy in herd immunity scenario.

Notice that the value $x_i^c(t)$ is arbitrarily selected on the basis of a condition vector (S) from every contaminated case x^c, so that $c = \{i|S(i) = 1\}$.

2. Susceptible case

The new gene value of $x_i^j(t+1)$ is influenced by any social gap within the spectrum of $r \in [\frac{1}{3}BRr, \frac{2}{3}BRr]$, which is determined by the discrepancy between the present gene and a gene extracted from a compromised case x^m, such as

$$x_i^j(t+1) = N(x_i^j(t)) \tag{6}$$

where

$$N\left(x_i^j(t)\right) = x_i^j(t) + r \times (x_i^j(t) - x_i^m(t)) \tag{7}$$

Notice that the value $x_i^m(t)$ is distributed from every resistant case x^m randomly, and that it is centered on a vector of status (S) given that $m = \{i|\ S(i) = 0\}$.

3. Immune case

The new gene value of $x_i^j(t+1)$ is influenced by any social gap within the spectrum of $r \in [\frac{2}{3}BRr, BRr]$, which is determined by the discrepancy between the present gene and a gene extracted from a compromised case x^v, such as

$$x_i^j(t+1) = R(x_i^j(t)) \tag{8}$$

where

$$R\left(x_i^j(t)\right) = x_i^j(t) + r \times (x_i^j(t) - x_i^v(t)) \tag{9}$$

Notice that the value $x_i^v(t)$ is distributed from every resistant case x^v randomly, and that it is centered on a vector of status (S) given that $f(x_i^v) = arg\ \min_{j\{k|S(k)=2\}} f(x_i^j)$.

Step 4: Update herd immunity population

The immunity rate $f(x^j(t+1))$ of each case $x^j(t+1)$ generated is determined and the actual case $x^j(t)$ is replaced by the obtained case $x^j(t+1)$ if the obtained case is stronger, such that $f(x^j(t+1)) < f(x^j(t))$. Also, the age vector Aj is increased by a value of 1 if $Sj = 1$. For each event, the state vector (Sj) is modified x^j based on the herd immune criterion that uses the following equation:

$$\xrightarrow{Sj} \begin{cases} 1 & f\left(x^j(t+1)\right) \\ & < \dfrac{f\left((x)^j(t+1)\right)}{\Delta f(x)} \wedge Sj = 0 \wedge is\_corona(x^j(t+1)) \\ 2 & f\left(x^j(t+1)\right) < \dfrac{f\left((x)^j(t+1)\right)}{\Delta f(x)} \wedge Sj = 1 \end{cases} \tag{10}$$

where the binary value of is_corona $(x^j(t+1))$ is equal to 1 when the new value is a value from any infected case that has been inherited by case $x^j(t+1)$. Also, the $\Delta f(x)$ is the mean significance of the immune population rates such as $\frac{\sum_{x_i}^{HIS} f(x_i)}{HIS}$.

Notice that the immunity levels of the individuals in the population are altered depending on the social gap measured earlier. If the newly produced individual immunity rate is better than the population's average immunity rate, this means that the population is becoming more immune to the virus. If the recently discovered population is sufficiently strong to be immune to the virus, then the threshold of herd immunity has been reached.

Phase 5: Fatal cases

In this phase, if the immunity rate $f(x^j(t+1))$ of the current infected case (Sj==1) cannot be strengthened as defined by the *Max_Age* parameter (i.e., $Aj >= Max\_Age$), then this case is considered dead. However, using $x_i^j(t+1) = Lbi + (Ubi - Lbi) \times U(0, 1)$, $\forall i = 1, 2, .., n$ is then regenerated from scratch. In addition, Aj and Sj are both set to 0. This phase may be beneficial in diversifying the current population and thereby avoiding local optima.

Phase 6: Stop criterion

The CHIO algorithm repeats step 3 to step 5 until the termination criterion is reached, which normally depends on whether

the maximum number of iterations is reached. In this case, the population is dominated by the total number of susceptible and immunized cases. Also, the infected cases are passed.

All the above phases of the CHIO algorithm are illustrated as a flowchart in Fig. 2.

The pseudocode of the six CHIO phases is given below:

CHIO algorithm pseudocode

1. **Step 1: Initialize the CHIO parameters** (*HIS*, 2. *Max_itr*, *and Max_Age*)
2. **Step 2: Generate herd immunity population**
3. $x_i^j = Lbi + (Ubi - Lbi) \times U(0, 1)$, $\forall i = 1, 2, .., n$ and $\forall j = 1, 2, .., HIS$
4. calculate the fitness of each search agent
5. set $Sj=0$ $\forall j = 1, 2, .., HIS$
6. set $Aj=0$ $\forall j = 1, 2, .., HIS$
7. **Step 3: Herd immunity evolution**
8. while $(t \leq Max\_itr)$ do
9. for $j = 1$ to HIS do
10. is $Corona(x^j\ (t + 1)$ = false
11. for $i = 1$ to N do
12. if $(r < 1/3 \times BRr)$ then
13. $x_i^j\ (t + 1) = C(x_i^j\ (t))$
14. is $Corona(x^j\ (t + 1)$ = true
15. else if $(r < 2/3 \times BRr)$ then
16. $x_i^j\ (t + 1) = N(x_i^j\ (t))$
17. else if $(r < BRr)$ then
18. $x_i^j\ (t + 1) = R(x_i^j\ (t))$
19. else
20. $x_i^j\ (t + 1) = x_i^j\ (t)$
21. end if
22. end for
23. **Step 4: Update herd immunity population**
24. if $f(x^j\ (t + 1)) < f(x^j\ (t))$ then
25. $f(x^j\ (t)) = f(x^j\ (t + 1))$
26. *else*
27. $Aj = Aj + 1$
28. end if
29. if $f\left(x^j\ (t + 1)\right) < \frac{f\left((x)^j(t+1)\right)}{\delta f(x)} Sj = 0 is\_corona(x^j\ (t + 1)$ then
30. *Sj=1, Aj=1*
31. end if
32. If $f\left(x^j\ (t + 1)\right) < \frac{f\left((x)^j(t+1)\right)}{\delta f(x)} Sj = 1$then
33. *Sj=2, Aj=0*
34. end if
35. **Step 5: Fatality condition**
36. if $Aj >= Max\_Age$ and $S==1$ then
37. $x_i^j = Lbi + (Ubi - Lbi) \times U(0, 1)$, $\forall i = 1, 2, .., n$
38. $Aj=0$
39. $Sj=0$
40. end if
41. end for
42. $t = t + 1$
43. end while

4. Greedy crossover (GC) operator

One of the essential search operators is the crossover operator. The main purpose of using this operator is to generate a new promising optimal solution by merging current parent solutions [87]. This is seen as an effective technique as the search process will then theoretically lead to new exploration regions where better solutions can be sought [88]. The crossover operator takes two solutions and combines them in order to create a new one that is distinct from the previous solutions because it selects the best features of both solutions to form an optimal solution. There are several forms of crossover operators, each of which relies on a particular mechanism [89]. In this study, a methodology based on a greedy approach proposed in [90] was used to find the best solutions in the crossover phase. Here, it was specifically implemented to enhance the exploration capability of the CHIO.

A greedy algorithm is a step-by-step approach as it ensures that the next step offers the maximum potential value on the route to a solution [91]. A greedy algorithm can also be used to solve optimization problems [80]. If better decisions can be made in any step and an optimal solution to the whole problem can be found, a greedy algorithm can ultimately eliminate the problem.

In this study, the application of a greedy strategy first involves the random selection of two parent solutions S_A and S_B from the population by CHIO, the first component of the offspring (S_O) apoints the values between S_A and S_B by performing an intersection process between the values, where $S_O = S_A \cap S_B$. Then, the remainder of the offspring is proceeded to a greedy strategy based on the potential P_x, which extends S_O step by step by assigning one element to it at each step until the offspring includes precisely N elements. In the following, the process of the greedy strategy is explained in more detail. Fig. 3 describes the idea of the GC strategy used in this study.

Let $N = \{S_1, S_2, \ldots, S_N\}$ be a set of elements, and let d_{XY} be the distance between them, where S_X and S_Y ($d_{XY} = d_{YX}$) elements with $d_{XY} > 0$ if $X \neq Y$ and $d_{XY} = 0$ otherwise. Then, potential P_x can be calculated, taking into consideration the objective function f in Eq. (13) in next section, as the following equation:

$$P_X = \sum_{S_Y \in S} d_{XY}, \quad S_x \in N \tag{11}$$

In the first step, all the elements in S_A are evaluated to define the element with the highest potential relative to S_O and move it from $S_A \backslash S_O$ to S_O. We then take into account the elements in $S_B \backslash S_O$, specify the element with the greatest potential in $S_B \backslash S_O$ and move it to S_O. Then, at each point of this greedy process, we consider the elements in $S_A \backslash S_O$ and $S_B \backslash S_O$ in turn until S_O reaches the size of N. The offspring in S_O is usually a relatively high-quality solution.

The features are swapped into the offspring in specific steps. Let swap (S_x, S_Y) define a transfer that switches $S_X \in S$ and $S_Y \in N \backslash S$. Then, as swap (S_x, S_Y) is used the target variation [90], which can be easily measured by:

$$Target\ Variation = f\left(S'\right) - f\left(S\right) = P_Y - P_x - d_{XY} \tag{12}$$

where $S' = S \backslash \{S_x\} \cup \{S_Y\}$ while P_Y and P_x are, respectively, the potential of S_Y, S_x according to Eq. (15).

The population updating process determines whether to come up with a solution for offspring, which is the crossover should be a part of the population and, if not, the current population solution will be substituted. Population management is a critical issue because updating the population rule actually influences the diversity of the population, which in turn affects the convergence of the CHIO search. Algorithm 2 provides the pseudocode of the GC operator.

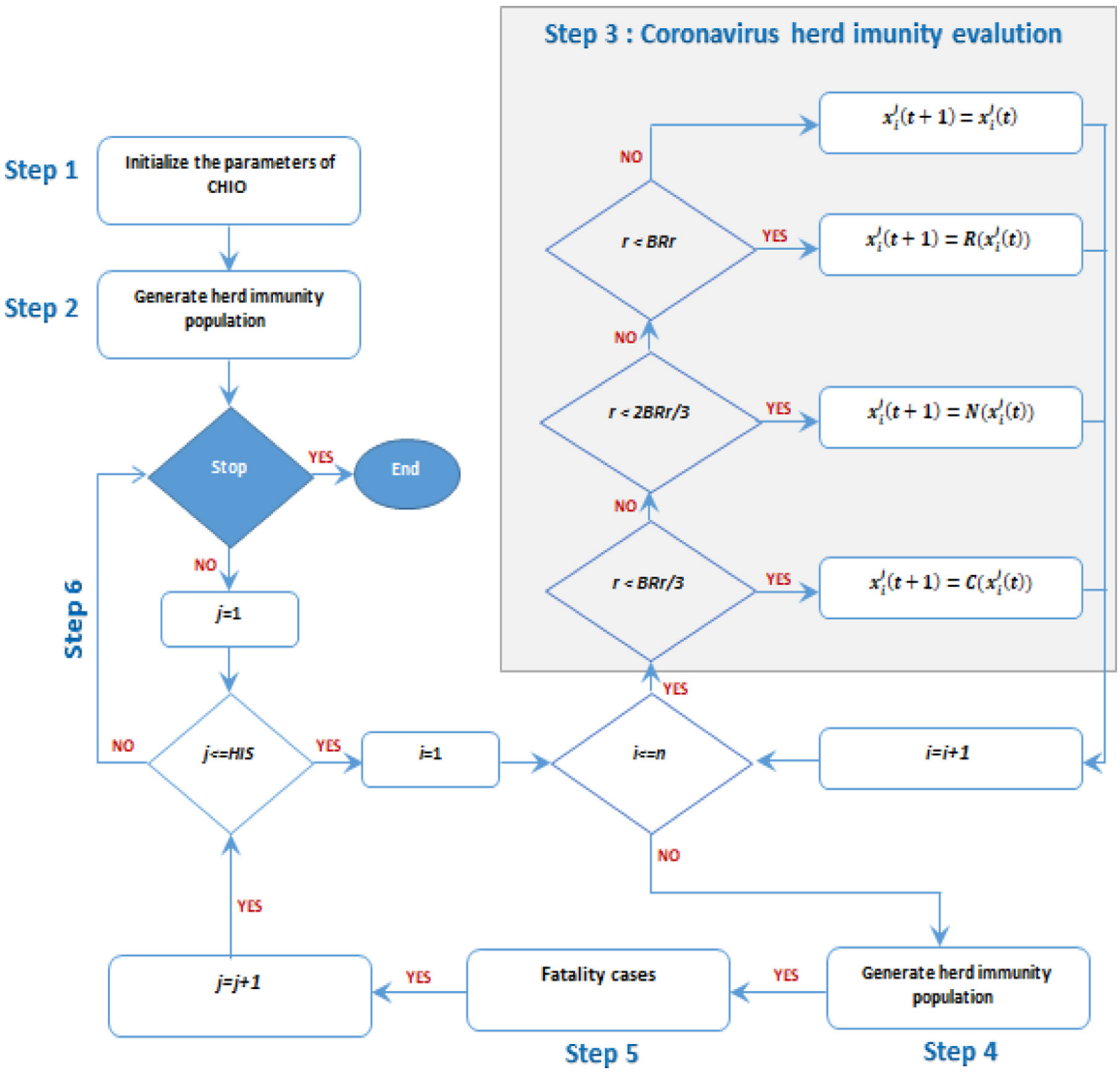

Fig. 2. Flowchart of CHIO algorithm.

Algorithm 2: Greedy crossover operator pseudocode

1.CHIO generalizes two parent solutions S_A and S_B
2.Greedy crossover: One offspring solution S_O:
3. $S_O \leftarrow S_A \cap S_B$ /* Establish a partial solution first by retaining mutual common features by S_A *and* S_B */
4. while $|S_O| < N$ do
5.Select from $S_A \setminus S_O$ the element u with the highest potential with respect to S_O
6.$S_O \leftarrow S_O \cup \{u\}$, $S_A \leftarrow S_A \setminus \{u\}$
7.if $|S_O| = N$ then
8.Return S_O and Stop
9.end if
10.Select from $S_B \setminus S_O$ the element v with the highest potential with respect to S_O
11.$S_O \leftarrow S_O \cup \{v\}$, $S_B \leftarrow S_B \setminus \{v\}$
12.end while
13.Return S_O

5. CHIO-GC for the FS problem

The FS process is used to delete redundant, obsolete and misleading features in order to obtain the best subset that represents the best outcome, where every feature is relevant if the choice depends on it, otherwise it is irrelevant. A feature is considered to be redundant if it is heavily associated with other features. The FS process is a binary optimization problem where solutions are limited to binary values (0, 1). This implies that any optimization strategy used to solve FS problems needs to be built in binary form, such that solutions are represented as either 0 or 1 in a one-dimensional vector. A feature that is selected is assigned the value 1, otherwise 0. Fig. 4 gives an example of the binary representation of selected features.

In this study, the wrapper FS approach is based on KNN, which determines the accuracy rate of the proposed approaches, CHIO and CHIO-GC, for the FS process in medical diagnosis. According to related works, the KNN classifier has been found to have good classification efficiency when applied to FS problems [4]. In this study, the number of nearest neighbors (K) was five. The 5-NN algorithm was used for the fitness assessment during the training period with internal N-fold cross-validation, where the number of folds was five; the average error rate in the classification procedure was determined for each fold of each equivalent particle. The number of folds (N) and the number of nearest neighbors (K) were chosen based on previous research.

The wrapper FS approach involved a generation, evaluation, a termination criterion and validation phase. In the wrapper

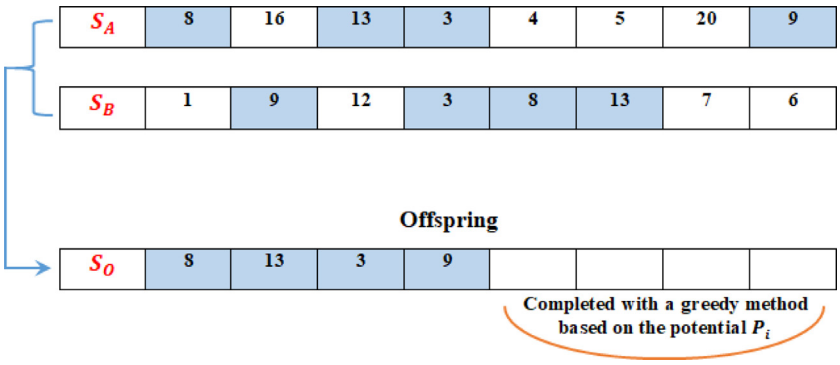

Fig. 3. Proposed greedy crossover operator.

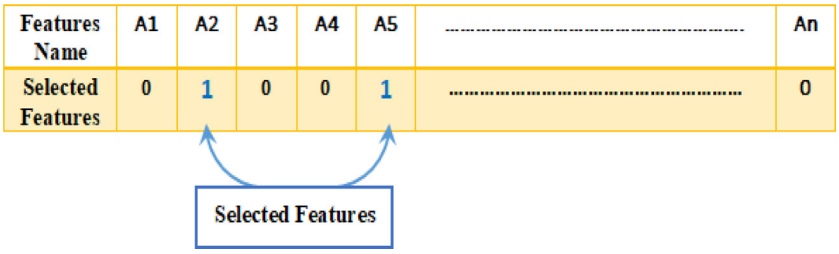

Fig. 4. Binary representation of feature selection.

approach, the learning algorithm was used as a part of the evaluation phase. This approach created an interaction between the search subset and the classification algorithm. Therefore the KNN classifier was used twice, one time in the evaluation and one time in the validation phase. During the generation step, a subset of features was chosen from the full dataset for the validation process to decide whether or not it matched the solution. In this phase, the exploration and exploitation capabilities of the CHIO algorithm were used to search for and generate a subset from a given dataset. To increase the exploration efficiency of the CHIO to the maximum possible degree, a crossover operator was used in a greedy manner to find the best solutions before CHIO moved on to the exploitation part of the search process. Fig. 5 describes the CHIO with GC for FS based on a wrapper approach using KNN.

The efficiency of the suggested approaches CHIO and CHIO-GC were assessed according to accuracy, recall, precision, F-measure and the number of features (selection size).

Classification accuracy is calculated by:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{13}$$

where:

True positives (TPs) are cases in which the model predicted true and the actual output was also true.

True negatives (TNs) are cases in which the model predicted false and the actual output was false.

False positives (FPs) are cases in which the model predicted true and the actual output was false.

False negatives (FNs) are cases in which the model predicted false and the actual output was true.

Precision describes how accurate the learning model is in terms of how many of the cases that the model predicted as positive are actually positive. Precision is calculated by:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{14}$$

Recall represents how many of the actual positives the model captures by labeling them as positive (true positive). Recall is calculated by:

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{15}$$

The F-measure expresses the balance between the ratio of recall and precision; the closer it is the higher and close to the degree of accuracy. The F-measure is calculated by:

$$\text{F} - \text{measure} = 2\frac{(\text{precision} \,.\, \text{recall})}{\text{precision} + \text{recall}} \tag{16}$$

6. Experimental results

This section describes the experimental setup and presents the analysis of the results, as well as comparisons with previous methods in order to assess the performance of the proposed method. The instability that has been generated depends on a variety of criteria, including the accuracy rate, the rate of convergence and certain measurements of central inclination. In order to conduct a fair scientific analysis, similar work environments and conditions were observed throughout the experiments. The experiments were carried out using an Intel® Core™ i7-6006U Processor @ 2.00 GHz (four CPUs), ~2.0 GHz with 8 GB of RAM. The CHIO was introduced using Matlab R2016a. The datasets were divided into 70% for training and 30% for testing. The tests were carried out over 30 runs for each dataset and 100 iterations were used in each run.

6.1. Parameter settings

In the experiments, the input parameters were determined by the results of some initial tests which enabled the proposed method to produce better output. In order for the results of the experiment to be the same, the algorithm configurations were identical throughout. Table 1 provides the CHIO parameter values that were used in all the experiments.

As for the KNN classifier, the input was the nearest training instances in the feature space and the output was a class membership. The labeling method depended on the majority of votes

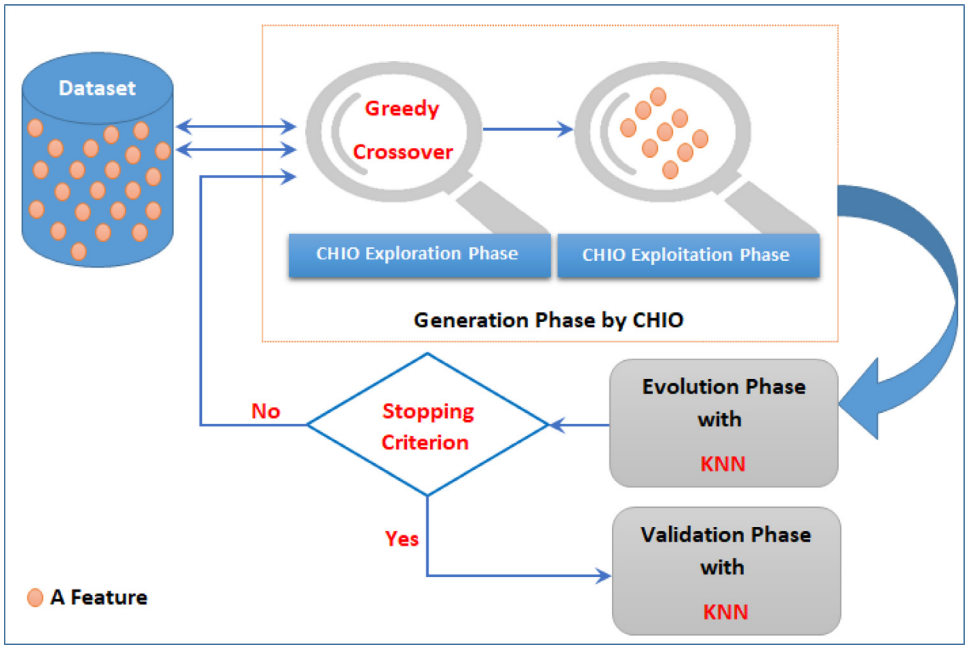

Fig. 5. Proposed CHIO-GC approach.

Table 1
Parameter settings.

Parameter	Value
HIS	30
Max_Age	100
BRr	0.01
Max_Itr	100
LB (lower bound)	0
UB (upper bound)	1

cast by the nearest K neighbors to the query. In the experiments, the average error rate findings indicated the output uncertainty of the KNN classifier when various values of K were used. The minimal error rate was reached when K = 5.

6.2. Description of the datasets

The term medical data refers to health-related data that are used to determine routine patient treatment or as part of a diagnostic trial program. There are several categories of such data, such as administrative data, claims data, patient disease data and clinical trial data, among others. In this study, in order to be able to generalize the results of the experiments, we used two types of medical data: a number of medical benchmark datasets and one real-world COVID-19 dataset.

6.2.1. Medical benchmark datasets

Twenty-three well-known benchmarked datasets of diverse patient data were used in the experiments. They were downloaded from a range of data repositories such as UCI, KEEL, and Kaggle, as well as other well-known websites for FS medical

Table 2
Characteristics of the medical benchmark datasets.

Dataset		Number of features	Number of instances	Number of classes	Source of dataset
1	Diagnostic	30	569	2	UCI
2	Original	9	699	2	UCI
3	Prognostic	33	194	2	UCI
4	Coimbra	9	115	2	UCI
5	BreastEW	30	596	2	UCI
6	Retinopathy	19	1151	2	UCI
7	Dermatology	34	366	6	UCI
8	ILPD-Liver	10	583	2	UCI
9	Lymphography	18	148	4	UCI
10	Parkinsons	22	194	2	UCI
11	ParkinsonC	753	755	2	UCI
12	SPECT	22	267	2	KEEL
13	Cleveland	13	297	5	KEEL
14	HeartEW	13	270	2	KEEL
15	Hepatitis	18	79	2	KEEL
16	SAHear	9	461	2	KEEL
17	Spectfheart	43	266	2	KEEL
18	Thyroid0387	21	7200	3	KEEL
19	Heart	13	302	5	Kaggle
20	Pima-diabetes	9	768	2	Kaggle
21	Leukemia	7129	72	2	https://jundongl.github.io/scikit-feature/datasets.html
22	Colon	2000	62	2	https://jundongl.github.io/scikit-feature/datasets.html
23	Prostate_GE	5966	102	2	https://jundongl.github.io/scikit-feature/datasets.html

Table 3
Features of the COVID-19 dataset.

Feature name		Description
1	id	Patient identifier
2	location	Patient location (local address)
3	country	Country of origin of the patient
4	gender	Gender of the patient
5	age	Age of the patient
6	sym_on	Date the patient shows symptoms
7	hosp_vis	Date the patient visits hospital
8	vis_wuhan	The patient has visited Wuhan
9	from_wuhan	The patient is from Wuhan
10	symptom1	A symptom presented by the patient
11	symptom2	A symptom presented by the patient
12	symptom3	A symptom presented by the patient
13	symptom4	A symptom presented by the patient
14	symptom5	A symptom presented by the patient
15	symptom6	A symptom presented by the patient

datasets. The characteristics of these datasets are detailed in Table 2.

As can be seen from Table 2, the 23 datasets include many case studies on medical diagnosis and have different structures. The power and reliability of the CHIO and CHIO-GC can be discovered by studying the improved optimizer on different problems with different characteristics. The datasets were divided into two parts: 70% for training and 30% for testing. The tests were carried out over 30 runs for each dataset and 100 iterations were used in each run.

6.2.2. COVID-19 dataset – a real-world dataset

In March 2020, the World Health Organization confirmed that extreme acute respiratory syndrome coronavirus 2 (SARS-CoV-2) or COVID-19, which had emerged in China in late 2019, had reached pandemic status. At the time of writing it has resulted in the death of millions of people worldwide. Artificial intelligence is increasingly being used in a range of technologies for diagnosis, identification and prevention in the global fight against COVID-19. Hence it seemed appropriate that the CHIO and CHIO-GC were applied to predict the health of COVID-19 patients. For this purpose, a real-world dataset on COVID-19 patients was obtained from https://github.com/AtharvaPeshkar/Covid-19-Patient-Health-Analytics. The dataset consisted of 15 features, as listed in Table 3.

In this study, patient data containing missing values for both "death" and "recovery" were excluded from the key dataset. For the experiments, the data were split evenly into two sets of training and testing data for the evaluation process.

7. Results and discussion

In order to evaluate the efficacy of the proposed approaches, CHIO and CHIO-GC, seven outcomes were taken into account: accuracy, error rate, number of features chosen (selection size), precision, recall, F-measure, boxplot and convergence speed.

First the CHIO and CHIO-GC were compared in terms of accuracy rate and selection size. The results achieved by the two approaches when applied to each of the 23 datasets and the COVID-19 dataset after 30 runs are provided in Table 4.

As can be seen from Table 4, CHIO-GC achieved higher accuracy in all datasets. This suggests that the CHIO is capable of generating more reliable results if its search mechanism is modified. It can also be seen from the table that the CHIO-GC approach was able to reduce the gap between the min and max accuracy values of the CHIO within 30 runs. In all datasets, the accuracy result was not less than that of the basic CHIO and, at the same time, the maximal values were part of the recommended CHIO-GC approach.

As for the FS size, the CHIO-GC demonstrated an advantage over the CHIO in terms of the number of features selected in 17 out of the 24 datasets, namely, Diagnostic, Original, Prognostic, Coimbra, BreastEW, ILPD-Liver, Lymphography, Parkinsons, ParkinsonC, SPECT, SAHear, Thyroid0387, Heart, Leukemia, Colon, Prostate_GE and COVID-19. These results demonstrate the power of the modification in the CHIO-GC to improve the exploration capability of the CHIO to find the best primitive solutions.

The precision, recall and F-measure values of the two approaches were also evaluated in order to further test the results and the extent of the classifier's ability to provide reliable, correlated and result values are equivalent in all sequences for each dataset. Precision represents the ratio of positive IDs that were actually right, while recall reflects the ratio of true positive IDs that were correctly detected, and the F-measure denotes the equilibrium between the recall and precision ratios. The precision, recall and F-measure values were determined by Eqs. (14), (15), and (16), respectively. The degree to which the efficiency of the CHIO and CHIO-GC approaches is adapted and concentrated in all of the datasets used in the experiment is shown in Table 5.

As shown in Table 5, all the F-measure values were higher than the accuracy values shown in Table 4 in all datasets except BreastEW and HeartEW. A large amount of real negatives, which in most technical situations are not relied upon, could have made a significant contribution to the results of the accuracy test. Even though FNs and FPs usually have market costs (quantifiable and non-quantifiable), the F-measure may be a better assessment to use where equilibrium between accuracy and recall is sought and the distribution of classes is inconsistent.

Precision, recall, and the F-measure give a more precise assessment of a classifier's behavior because they can be used to obtain a more in-depth judgment of the classifier's ability to find the correct results in learning than by assessing its performance based on accuracy alone. When the consequences of false positives are significant, precision becomes a useful assessment metric, whereas when the cost of false negatives is significant, recall can provide further insight into the results. Also, the F-measure is useful for understanding the tradeoff between accuracy and coverage when categorizing positive cases because it provides a more accurate estimate of wrongly categorized instances than the accuracy metric. The F-measure provides an overall assessment of a model's reliability that mixes precision and recall, in the same way as addition and multiplication can mix two components to produce a different result entirely. Hence, a strong F1 score indicates the presence of a small sample size of false positives and false negatives, thus false alarms do not affect the results.

In addition, the T-test was used to compare the efficiency of the CHIO and CHIO-GC approaches. Using these proposed approaches, which rely on the precision of the results specific to each dataset, the findings statistics are carried out. By conducting a T-test, with a 95% spectrum of significance (alpha = 0.05) on the p-values obtained and the classification accuracy the, different corresponding statistics are shown in Table 6.

As can be seen from Table 6, the efficiency of the CHIO-GC is slightly higher than that of the original CHIO, where most of the P-values for the 24 datasets are less than 0.0001. These findings show that the use of the CHIO-GC is effective for the solution of FS problems.

A boxplot is a charting technique for displaying a five-number summary. The interquartile range denotes the location of the data's middle part. The first quartile (the 25% mark) and third quartile (the 75% mark) are located at the respective ends of the box. The chart's lowest point is on the far left, while its maximum is on the far right. The median is indicated by a vertical bar in the center of the box. A boxplot indicates how closely grouped the data are and whether they are symmetrical. It also exposes

Table 4
Accuracy and feature selection size results for CHIO and CHIO-GC.

Dataset		Average accuracy		Max accuracy		Min accuracy		Selection size	
		CHIO	CHIO-GC	CHIO	CHIO-GC	CHIO	CHIO-GC	CHIO	CHIO-GC
1	Diagnostic	0.8540	0.9033	0.91	0.96	0.79	0.84	14.4000	13.3700
2	Original	0.9233	0.9710	0.96	0.99	0.85	0.94	6.2000	5.1040
3	Prognostic	0.5293	0.6716	0.62	0.77	0.47	0.60	16.2212	14.6202
4	Coimbra	0.8006	0.8896	0.87	0.91	0.70	0.86	4.6667	3.6007
5	BreastEW	0.8993	0.9400	0.94	0.97	0.85	0.89	15.8333	13.7303
6	Retinopathy	0.4660	0.6436	0.61	0.69	0.38	0.60	7.4667	7.2647
7	Dermatology	0.6690	0.8006	0.73	0.87	0.55	0.70	18.5000	18.4900
8	ILPD-Liver	0.6423	0.7716	0.69	0.79	0.60	0.72	4.1098	4.0000
9	Lymphography	0.7606	0.8343	0.82	0.91	0.69	0.79	10.1667	10.0622
10	Parkinsons	0.6690	0.7903	0.73	0.85	0.55	0.75	9.8333	9.7383
11	ParkinsonC	0.6856	0.8400	0.79	0.88	0.58	0.78	366.7333	365.8322
12	SPECT	0.6073	0.6960	0.68	0.88	0.55	0.60	9.7000	9.6050
13	Cleveland	0.4896	0.5966	0.58	0.64	0.44	0.55	6.7667	6.8097
14	HeartEW	0.8540	0.9116	0.91	0.94	0.79	0.87	6.4000	7.0105
15	Hepatitis	0.6690	0.7903	0.73	0.85	0.55	0.75	8.2000	8.2011
16	GCHear	0.6420	0.7036	0.70	0.73	0.59	0.68	4.5333	3.1551
17	Spectfheart	0.6716	0.7303	0.77	0.79	0.60	0.68	20.9333	21.0030
18	Thyroid0387	0.8986	0.9603	0.96	0.98	0.82	0.92	10.0314	8.0116
19	Heart	0.7316	0.8126	0.79	0.87	0.64	0.77	8.1000	6.1505
20	Pima-diabetes	0.7153	0.7956	0.86	0.87	0.61	0.68	5.4667	6.8387
21	Leukemia	0.9876	0.9900	1.0000	1.0000	0.91	0.93	3597.4427	3560.5107
22	Colon	0.6203	0.7176	0.70	0.82	0.55	0.60	1011.4927	1000.0067
23	Prostate_GE	0.4750	0.6010	0.59	0.64	0.39	0.55	3045.7317	2979.4116
COVID-19 dataset		0.9135	0.9370	0.9482	0.9770	0.8402	0.8818	4.1100	3.0500

Table 5
Precision, recall and F-measure results for CHIO and CHIO-GC.

Dataset		Precision		Recall		F-Measure	
		CHIO	CHIO-GC	CHIO	CHIO-GC	CHIO	CHIO-GC
1	Diagnostic	0.913978	0.954436	0.850000	0.954436	0.880829	0.954436
2	Original	0.945455	0.973568	0.945455	0.977876	0.945455	0.975717
3	Prognostic	0.689655	0.821333	0.714286	0.800000	0.701754	0.810526
4	Coimbra	0.825581	0.944444	0.855422	0.876289	0.840237	0.909091
5	BreastEW	0.893617	0.921875	0.840000	0.983333	0.865979	0.951613
6	Retinopathy	0.615385	0.676471	0.592593	0.718750	0.603774	0.696970
7	Dermatology	0.768571	0.925000	0.770774	0.787234	0.769671	0.850575
8	ILPD-Liver	0.750000	0.843854	0.677419	0.814103	0.711864	0.828711
9	Lymphography	0.790576	0.858586	0.848315	0.904255	0.818428	0.880829
10	Parkinsons	0.720000	0.860000	0.720000	0.924731	0.720000	0.891192
11	ParkinsonC	0.783784	0.900000	0.814607	0.964286	0.798898	0.931034
12	SPECT	0.666667	0.789041	0.615385	0.822857	0.640000	0.805594
13	Cleveland	0.578947	0.625000	0.687500	0.689655	0.628571	0.655738
14	HeartEW	0.889552	0.925969	0.892216	0.956311	0.890882	0.940896
15	Hepatitis	0.768675	0.829268	0.848404	0.855346	0.806574	0.842105
16	GCHear	0.604651	0.780488	0.896552	0.822857	0.722222	0.801113
17	Spectfheart	0.805797	0.808451	0.761644	0.803922	0.783099	0.806180
18	Thyroid0387	0.878505	0.983051	0.959184	0.974790	0.917073	0.978903
19	Heart	0.774026	0.858586	0.851429	0.885417	0.810884	0.871795
20	Pima-diabetes	0.769863	0.793689	0.831361	0.893443	0.799431	0.840617
21	Leukemia	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
22	Colon	0.555556	0.814016	0.740741	0.825137	0.634921	0.819539
23	Prostate_GE	0.555556	0.617647	0.689655	0.777778	0.615385	0.688525
COVID-19 dataset		0.912938	0.991736	0.907452	0.937500	0.910187	0.963855

the existence and coordinates of any outliers. Fig. 6 shows the boxplots that describe the distribution of the performance of CHIO and CHIO-GC when applied to the 24 datasets over 30 runs.

Note from Fig. 6 that the greedy approach was able to reduce the gap between the minimum and maximum accuracy values of the CHIO algorithm, and bring them closer to the value of the mean. Also, the minimum and maximum values of the CHIO-GC were higher than that of the basic algorithm. This is a clear indication of the ability of the CHIO-GC approach to improve the balance between exploration and exploitation of the CHIO search to produce accurate results.

Fig. 7 shows how the GC strategy was able to reduce the gap between the maximum and minimum accuracy values, and thus maximize accuracy. The accuracy values of the CHIO-GC approach were never worse than the CHIO values in all datasets, The results in the figure were arrived at by averaging the values of CHIO and CHIO-GC over the 30 runs for all datasets.

It is well known that a stable and rapid rate of convergence will lead to better solutions. Thus, in order to further test the efficiency of the CHIO and CHIO-GC, the convergence speed behavior curves of the two approaches were obtained by applying them to each of the 24 datasets over 30 different iterations. In general, any metaheuristic algorithm may take a large number of iterations to reach the optimum point. Therefore, it is important to employ methods that have a convergence rate that is as fast as possible. An algorithm's rate of convergence is usually measured by the number of iterations and by the number of function evaluations that are needed to obtain an acceptable solution.

The results in Fig. 8 indicate that the CHIO-GC was able to boost the global search of the original CHIO and thus improve

Table 6
T-test results for CHIO and CHIO-GC.

Dataset		Method	Mean	Std. deviation	Std. error mean	P-Value
1	Diagnostic	CHIO	0.8540	0.02673	0.00488	00.00
		CHIO-GC	0.9033	0.04046	0.00739	
2	Original	CHIO	0.9233	0.02523	0.00461	00.00
		CHIO-GC	0.9710	0.01062	0.00194	
3	Prognostic	CHIO	0.5293	0.04638	0.00847	00.00
		CHIO-GC	0.6716	0.04442	0.00811	
4	Coimbra	CHIO	0.8006	0.04548	0.00830	00.00
		CHIO-GC	0.8896	0.00999	0.00182	
5	BreastEW	CHIO	0.8993	0.02100	0.00383	00.00
		CHIO-GC	0.9400	0.01912	0.00349	
6	Retinopathy	CHIO	0.4660	0.06106	0.01115	00.00
		CHIO-GC	0.6436	0.02553	0.00466	
7	Dermatology	CHIO	0.6690	0.04088	0.00746	00.00
		CHIO-GC	0.8006	0.04548	0.00830	
8	ILPD-Liver	CHIO	0.6423	0.02609	0.00476	00.00
		CHIO-GC	0.7716	0.01744	0.00318	
9	Lymphography	CHIO	0.7606	0.03483	0.00636	00.00
		CHIO-GC	0.8343	0.02661	0.00486	
10	Parkinsons	CHIO	0.6690	0.04088	0.00746	00.00
		CHIO-GC	0.7903	0.01903	0.00347	
11	ParkinsonC	CHIO	0.6856	0.06611	0.01207	00.00
		CHIO-GC	0.8400	0.02197	0.00401	
12	SPECT	CHIO	0.6073	0.02840	0.00518	00.00
		CHIO-GC	0.6960	0.06667	0.01217	
13	Cleveland	CHIO	0.4896	0.04173	0.00762	00.00
		CHIO-GC	0.5966	0.02496	0.00456	
14	HeartEW	CHIO	0.8540	0.02673	0.00488	00.00
		CHIO-GC	0.9116	0.01783	0.00325	
15	Hepatitis	CHIO	0.6690	0.04088	0.00746	00.00
		CHIO-GC	0.7903	0.01903	0.00347	00.00
16	SAHear	CHIO	0.6420	0.03089	0.00564	00.00
		CHIO-GC	0.7036	0.01066	0.00195	
17	Spectfheart	CHIO	0.6716	0.04442	0.00811	00.00
		CHIO-GC	0.7303	0.03178	0.00580	
18	Thyroid0387	CHIO	0.8986	0.04455	0.00813	00.00
		CHIO-GC	0.9603	0.01474	0.00269	
19	Heart	CHIO	0.7316	0.04009	0.00732	00.00
		CHIO-GC	0.8126	0.02612	0.00477	
20	Pima-diabetes	CHIO	0.7153	0.05557	0.01015	00.00
		CHIO-GC	0.7956	0.03757	0.00686	
21	Leukemia	CHIO	0.9876	0.01736	0.00317	00.00
		CHIO-GC	0.9900	0.01017	0.00186	
22	Colon	CHIO	0.6203	0.03634	0.00663	00.00
		CHIO-GC	0.7176	0.05380	0.00982	
23	Prostate_GE	CHIO	0.4750	0.06329	0.01155	00.00
		CHIO-GC	0.6010	0.02537	0.00463	
24	COVID-19 dataset	CHIO	0.9135	0.02523	0.00461	00.00
		CHIO-GC	0.9370	0.01912	0.00349	

classification accuracy at a higher convergence speed compared to the CHIO. As can also be seen from Fig. 8, the CHIO-GC did not need to go beyond the 13th iteration to boost the solution in all datasets, except for the Danmini doorbell dataset. This finding confirms that the CHIO-GC has the potential to increase the speed of convergence.

7.1. Comparison with previous methods

The above results indicated that the CHIO-GC performed better than the original CHIO. In order to further assess the reliability of the CHIO-GC, and its ability to produce a high degree of classification accuracy while at the same time minimizing the number of attributes, it was compared with six methods in the literature. First, the CHIO-GC was compared with LBMFO-V3 [79] by using the 23 medical benchmark datasets. Then, it was compared against HLBDA [78] using the COVID-19 dataset. Finally, its performance was compared with that of four filter methods, namely, Chi-square, Relief, correlation-based feature selection (CFS) and information gain (IG).

7.1.1. Comparison with LBMFO-V3

The CHIO-GC was compared with LBMFO-V3 in terms of average classification accuracy and number of selected features using the 23 medical benchmark datasets. Table 7 shows the results of this comparison.

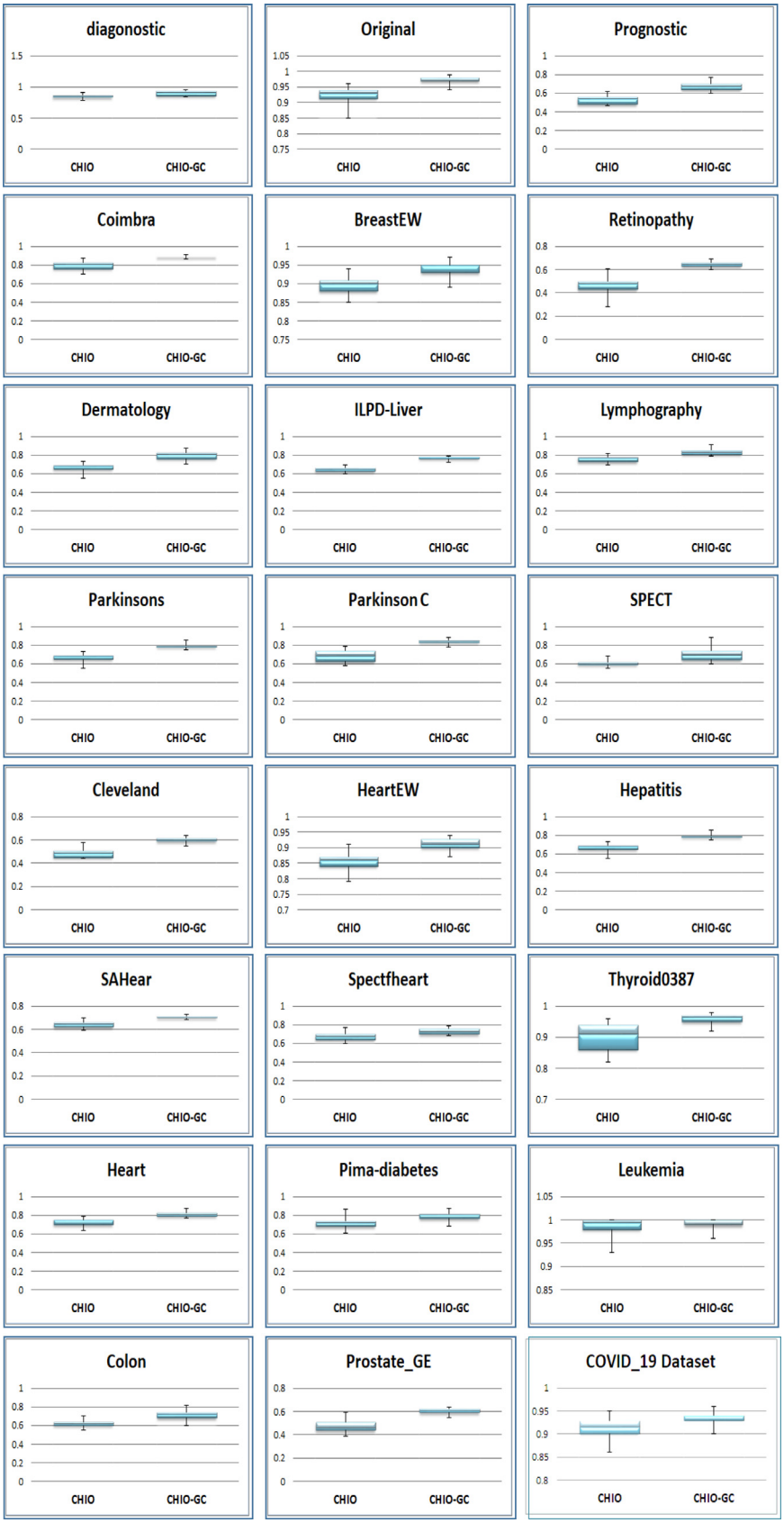

Fig. 6. Boxplots of CHIO and CHIO-GC for all datasets.

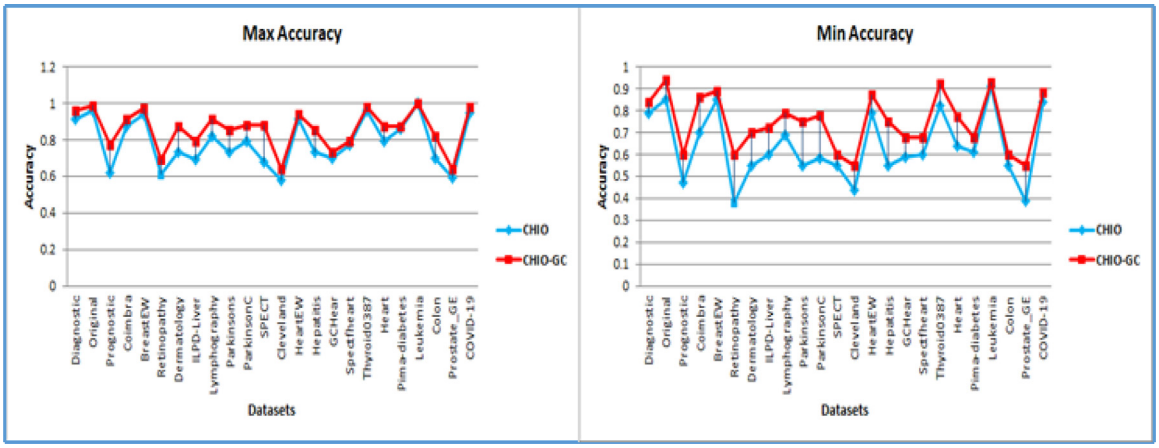

Fig. 7. Average of min and max accuracy of CHIO and CHIO-GC for all datasets.

Table 7
Comparison of CHIO-GC and LBMFO-V3.

Dataset		Average accuracy		Selection size	
		CHIO-GC	LBMFO-V3	CHIO-GC	LBMFO-V3
1	Diagnostic	0.9033	**0.9100**	**13.3700**	13.9991
2	Original	**0.9710**	0.9683	**5.1040**	5.5000
3	Prognostic	**0.6716**	0.5851	**14.6202**	15.0126
4	Coimbra	0.8896	**0.9312**	3.6007	**3.5103**
5	BreastEW	**0.9400**	0.9398	**13.7303**	13.9714
6	Retinopathy	**0.6436**	0.5380	7.2647	**6.9002**
7	Dermatology	0.8006	**0.8442**	18.4900	**18.3541**
8	ILPD-Liver	**0.7716**	0.7143	4.0000	4.0000
9	Lymphography	**0.8343**	0.8002	10.0622	**9.7520**
10	Parkinsons	**0.7903**	0.7689	**9.7383**	10.3584
11	ParkinsonC	**0.8400**	0.8190	**365.8322**	369.1070
12	SPECT	**0.6960**	0.6576	**9.6050**	10.7832
13	Cleveland	**0.5966**	0.5333	6.8097	**6.6899**
14	HeartEW	0.9116	**0.9388**	7.0105	**6.3100**
15	Hepatitis	**0.7903**	0.7500	**8.2011**	8.3569
16	SAHear	**0.7036**	0.6992	**3.1551**	3.2222
17	Spectfheart	**0.7303**	0.7013	21.0030	**20.4598**
18	Thyroid0387	0.9603	**0.9776**	**8.0116**	8.4563
19	Heart	**0.8126**	0.7603	**6.1505**	6.2752
20	Pima-diabetes	0.7956	**0.8065**	6.8387	**6.7612**
21	Leukemia	0.9900	**1.0000**	**3560.5107**	3570.7137
22	Colon	**0.7176**	0.6667	1000.0067	**991.5551**
23	Prostate_GE	**0.6010**	0.5056	**2979.4116**	2984.7153
COVID-19 dataset		**0.798321739**	0.774604348	**351.4142087**	351.9462565

Table 7 shows that the CHIO-GC outperformed LBMFO-V3 in terms of classification accuracy in 16 datasets, namely, Original, Prognostic, BreastEW, Retinopathy, ILPD-Liver, Lymphography, Parkinsons, ParkinsonC, SPECT, Cleveland, Hepatitis, SAHear, Spectfheart, Heart, Colon and Prostate_GE. The CHIO-GC approach had an overall accuracy of 0.7983 in all datasets, as compared to LBMFO-V3, which achieved 0.7746.

Moreover, the CHIO-GC also performed better than LBMFO-V3 in terms of selection size in 13 datasets, namely, Diagnostic, Original, Prognostic, BreastEW, Parkinsons, ParkinsonC, SPECT, Hepatitis, SAHear, Thyroid0387, Heart, Leukemia and Prostate_GE. The two approaches achieved the same result in one dataset, namely, ILPD-Liver. The CHIO-GC approach generated an overall selection size of 351.4142 features in all datasets, and it overcame the LBMFO-V3, which got 351.9462 features. Fig. 9 graphically illustrates the average accuracy and selection size achieved by CHIO-GC and LBMFO-V3 in all 23 datasets.

7.1.2. Comparison with HLBDA

The CHIO-GC was compared with the HLBDA in terms of classification accuracy and number of selected features using the COVID-19 dataset. In terms of classification accuracy the CHIO-GC had an 0.9370 average accuracy rate over 30 runs as compared to 0.9221 obtained by the HLBDA. On the other hand, the HLBDA generated an average of three features in all runs, whereas the CHIO-GC obtained a rate of four features. The features that were selected most frequently by the CHIO-GC across the 30 runs were location, country, age and symptom2. Fig. 10 shows the average accuracy and selection size results for the CHIO-GC and HLBDA using COVID-19 dataset.

7.1.3. Comparison with filter methods

The classification accuracy results of the CHIO-GC, which is a wrapper-based approach, were also compared against those of four general filter-based approaches, namely, Chi-square, relief, CFS and IG. These four filter methods offer a high assurance of the intensity values of the datasets. Although wrapper models require that the predictor is optimized as part of the selection phase, filter models focus on the general characteristics of the training data to choose features that are independent of some predictor. The filters used in this part of the analysis were accessed from the WEKA data mining program [92]. Table 8 shows the average

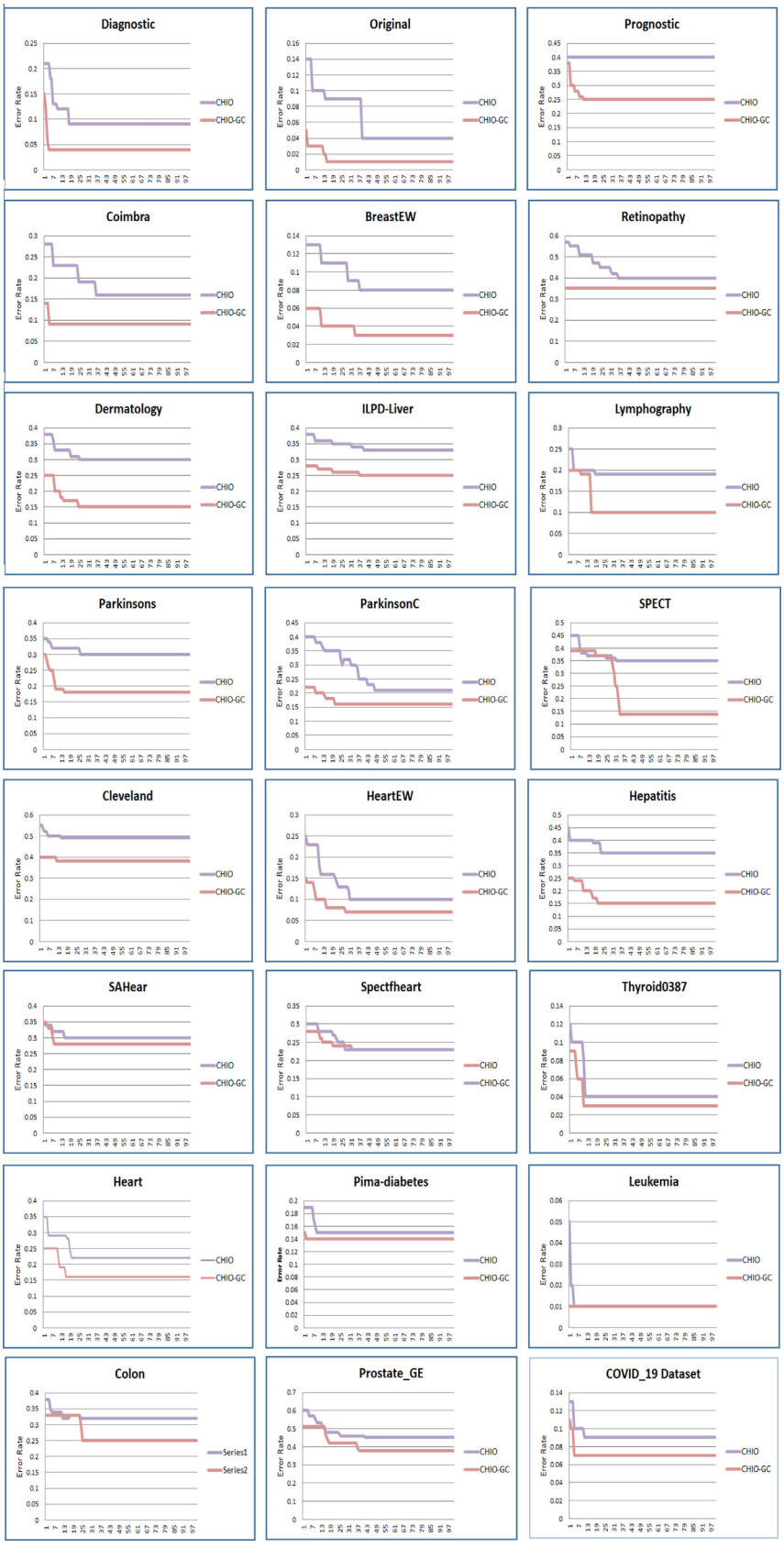

Fig. 8. Convergence speed of CHIO and CHIO-GC.

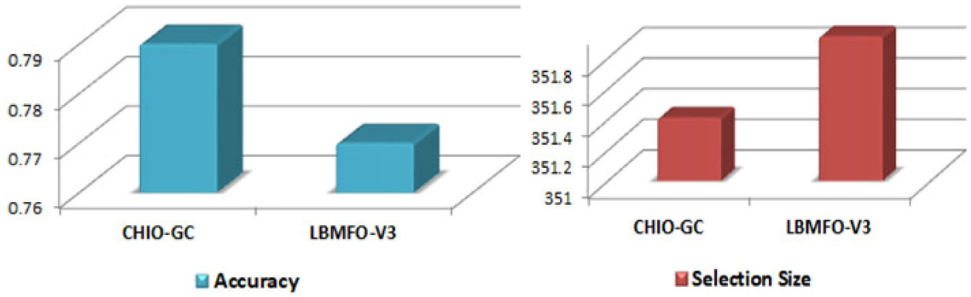

Fig. 9. Average accuracy and selection size of CHIO-GC and LBMFO-V3.

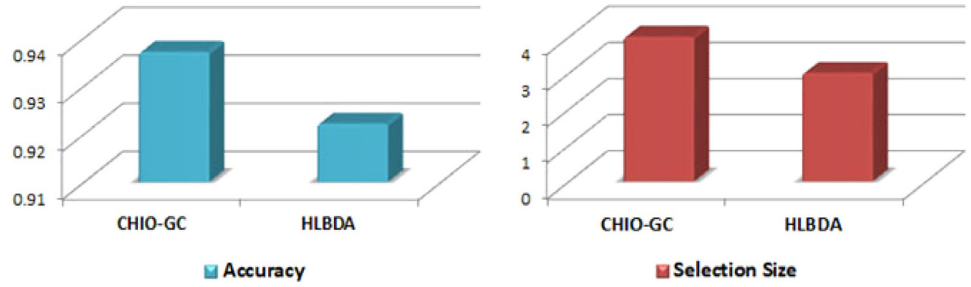

Fig. 10. Average accuracy and selection size of CHIO-GC and HLBDA.

Table 8
Average accuracy of CHIO-GC and filter methods.

Dataset		CHIO-GC	Chi-square	Relief	CFS	IG
1	Diagnostic	0.9033	0.5714	0.9585	**0.9533**	0.9349
2	Original	**0.9710**	0.9091	0.6426	0.6860	0.6759
3	Prognostic	0.6716	0.5910	**0.7727**	0.7576	0.7577
4	Coimbra	**0.8896**	0.3846	0.6672	0.5763	0.5578
5	BreastEW	**0.9400**	0.9365	0.8160	0.8029	0.8128
6	Retinopathy	**0.6436**	0.6349	0.5036	0.4783	0.5393
7	Dermatology	**0.8006**	0.7250	0.7248	0.4732	0.4021
8	ILPD-Liver	**0.7716**	0.7106	0.5119	0.5223	0.5264
9	Lymphography	0.8343	**0.8824**	0.5886	0.5533	0.5204
10	Parkinsons	**0.7903**	0.7581	0.7588	0.7360	0.7150
11	ParkinsonC	**0.8400**	0.6593	0.6590	0.6487	0.6376
12	SPECT	0.6960	**0.9667**	0.5651	0.5508	0.5460
13	Cleveland	**0.5966**	0.3940	0.1181	0.0398	0.0826
14	HeartEW	0.9116	**0.9334**	0.6153	0.5757	0.6202
15	Hepatitis	**0.7903**	0.7778	0.5538	0.5857	0.6417
16	SAHear	**0.7036**	0.6471	0.5024	0.5115	0.5227
17	Spectfheart	**0.7303**	0.7000	0.6079	0.6279	0.5551
18	Thyroid0387	0.9603	**1.0000**	0.6379	0.6955	0.9773
19	Heart	**0.8126**	0.5333	0.6317	0.5575	0.6114
20	Pima-diabetes	**0.7956**	0.6905	0.5147	0.5426	0.5264
21	Leukemia	**0.9900**	0.7120	0.6883	0.6759	0.6410
22	Colon	**0.7176**	0.5850	0.5641	0.5116	0.5097
23	Prostate_GE	**0.6010**	0.5042	0.5033	0.4786	0.4421
Average		**0.7983**	0.7046	0.6133	0.5887	0.5981

accuracy achieved by the CHIO-GC wrapper approach and by the four filter methods after applying them 30 times to the 23 medical benchmark datasets.

It can be observed from Table 8 that the CHIO-GC exceeded IG in all datasets. It also exceeded CFS and Relief in all datasets except Diagnostic and Prognostic, respectively. On the other hand, Chi-square performed better than the three other filter methods, and surpassed the results produced by the CHIO-GC in four datasets, namely, Lymphography, SPECT, HeartEW and Thyroid0387. However, overall, the CHIO-GC defeated all the filter methods in 17 datasets with an accuracy rate of 0.7983. Fig. 11 shows the accuracy rate of the CHIO-GC and the four filter methods.

7.2. Discussion

The results produced by the original CHIO indicate that the algorithm has a suitable balance between exploration and exploitation in its search mechanism. This balance is one of the most important strengths of metaheuristic algorithms as it helps them to find the best solutions during the search process. In the CHIO-GC approach, the exploration capability of the CHIO was modified by applying a greedy crossover operator to select the initial features with the aim of maximizing the solution. This modification contributed to an enhancement of the balance between exploration and exploitation. The stronger balance that was achieved also enabled the CHIO-GC to accelerate the rate of convergence during its search for the best solutions. The selection of the most appropriate features ensures that the identified solutions converge to the maximum. Thus exploration by means of randomization facilitates the search of the solution space from a local point of view and at the same time increases the variety of solutions.

The success of the proposed CHIO-GC approach in achieving a good balance between exploration and exploitation was demonstrated in the experiments in several ways. First, the CHIO-GC outperformed the CHIO in all 24 datasets in terms of classification accuracy, as shown in Table 4. Furthermore, the superiority of the CHIO-GC was also observed in the maximum and minimum accuracy it was able to achieve in each run, as shown in Table 4, Figs. 6 and 7. The CHIO-GC was able to minimize the gap between the maximum and the minimum accuracy and make them converge. Also, the convergence speed results showed the power of modify the exploration search which is made up of two criteria, the number of iteration needed to get the optimal solution and the initial started point of solution. So, As shown in Fig. 8, the CHIO-GC was better than the CHIO in terms of convergence speed as it had a good initial starting point and it did not need more than 30 iterations to obtain the optimal solution in most datasets.

Moreover, the CHIO-GC outperformed LBMFO-V3 in 16 datasets and HLBDA in the COVID-19 dataset in terms of both classification accuracy and selection size. Hence the inclusion of GC in the CHIO approach was proved to be beneficial in correcting the suboptimal solutions that were reached at premature

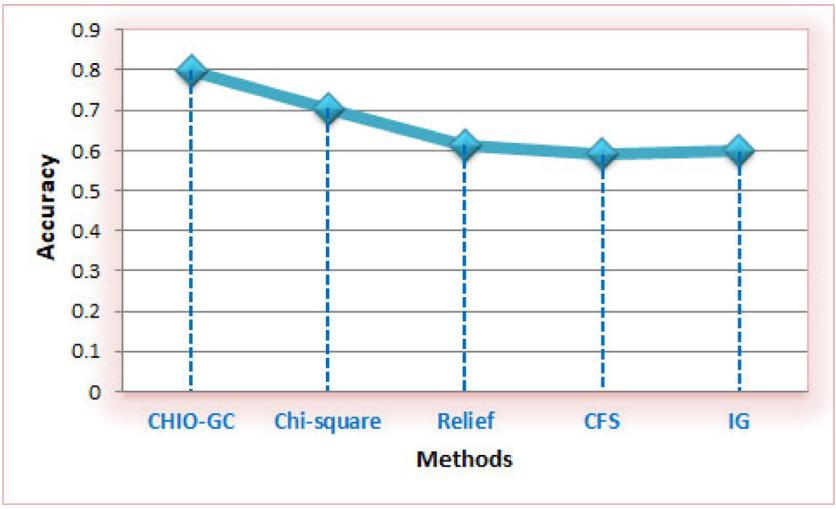

Fig. 11. Accuracy rate of CHIO-GC and filter approaches.

convergence and when trapped in a suboptimal search space. Furthermore, the CHIO-GC demonstrated its superiority when applied to large datasets such as ParkinsonC, Leukemia, Colon and Prostate_GE. Therefore, the proposed the CHIO-GC approach can be relied upon to solve FS problems if the problem becomes larger.

8. Conclusion

The FS problem is among the most critical issues facing researchers in many fields including medical diagnosis. In recent years, metaheuristics have been commonly used for FS to try to minimize the number of features required to achieve sufficiently reliable results, with the goal of increasing reliability and enhancing performance. In this study, a new metaheuristic named the coronavirus herd immunity optimizer (CHIO) was implemented to solve FS problems in medical diagnosis.

The CHIO was applied as a basic algorithm and as a modified algorithm using greedy crossover (CHIO-GC) to enhance exploration. Two types of dataset were used to assess the proposed approaches: 23 medical benchmark datasets and a real-world COVID-19 dataset. The evaluation of the two approaches was conducted in respect of several criteria, including classification accuracy, number of selected features, error rate, precision, recall, F-measure, boxplot, convergence speed and T-test. All the obtained results indicated that the CHIO-GC enhanced the exploration capability of the original CHIO.

In comparison experiments, the CHIO-GC outperformed two FS wrapper approaches, LBMFO-V3 and the HLBDA. The CHIO-GC surpassed LBMFO-V3 in 16 out of the 23 medical benchmark datasets with an accuracy rate of 0.79 and selection size rate of 351 features. It also outdid the HLBDA when applied to the COVID-19 dataset, with a classification accuracy of 0.93. Furthermore, the wrapper-based CHIO-GC surpassed four filter methods, namely, Chi-square, Relief, CFS and IG.

It is considered that these promising results were achieved through the strong balance between the two search phases of the CHIO-GC during the discovery of the right solutions, which also accelerated the convergence rate. This was achieved by incorporating a greedy crossover method into the CHIO algorithm to correct the suboptimal solution reached on premature convergence and when trapped in a local optimum search space.

In future work, researchers may wish to consider hybridizing the CHIO with another single-based metaheuristic algorithm such as simulated annealing to try to enhance its exploitation (local) search capability, or applying it in another FS field such as intrusion detection or image segmentation. Furthermore, many other computational intelligence algorithms can be used to solve medical diagnosis problems, such as monarch butterfly optimization (MBO), the earthworm optimization algorithm (EWA), elephant herding optimization (EHO), the moth search (MS) algorithm, the slime mold algorithm (SMA), and Harris hawks optimization (HHO).

CRediT authorship contribution statement

Mohammed Alweshah: Design and implementation of the research, Analysis of the results, Writing of the manuscript. **Saleh Alkhalaileh:** Design and implementation of the research, Analysis of the results, Writing of the manuscript. **Mohammed Azmi Al-Betar:** Design and implementation of the research, Analysis of the results, Writing of the manuscript. **Azuraliza Abu Bakar:** Design and implementation of the research, Analysis of the results, Writing of the manuscript.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The research reported in this publication was supported by the Deanship of Scientific Research and Innovation at Al-Balqa Applied University in Jordan. Grant Number: DSR-2019#144.

References

[1] W. Li, G.-G. Wang, A.H. Gandomi, A survey of learning-based intelligent optimization algorithms, Arch. Comput. Methods Eng. (2021) 1–19.
[2] S.-H. Liew, Y.-H. Choo, Y.F. Low, Fuzzy-rough classification for brainprint authentication, Jordan. J. Comput. Inf. Technol. (JJCIT) 5 (02) (2019).
[3] K. Dyczkowski, Intelligent medical decision support system based on imperfect information, in: The Case of Ovarian Tumor Diagnosis, Studies in Computational Intelligence, Springer, 2018.
[4] M. Alweshah, S. Alkhalaileh, D. Albashish, M. Mafarja, Q. Bsoul, O. Dorgham, A hybrid mine blast algorithm for feature selection problems, Soft Comput. 25 (1) (2021) 517–534.
[5] S. Sengan, G. Kamalam, J. Vellingiri, J. Gopal, P. Velayutham, V. Subramaniyaswamy, Medical information retrieval systems for e-health care records using fuzzy based machine learning model, Microprocess. Microsyst. (2020) 103344.

[6] A.R. Mishra, P. Rani, R. Krishankumar, K. Ravichandran, S. Kar, An extended fuzzy decision-making framework using hesitant fuzzy sets for the drug selection to treat the mild symptoms of Coronavirus Disease 2019 (COVID-19), Appl. Soft Comput. 103 (2021) 107155.

[7] H.E. Kiziloz, Classifier ensemble methods in feature selection, Neurocomputing 419 (2021) 97–107.

[8] J.F. Hair Jr., M. Sarstedt, Data, measurement, and causal inferences in machine learning: opportunities and challenges for marketing, J. Mark. Theory Pract. (2021) 1–13.

[9] A.M. Alshareef, A.A. Bakar, A.R. Hamdan, S.M.S. Abdullah, M. Alweshah, A case-based reasoning approach for pattern detection in Malaysia rainfall data, Int. J. Big Data Intell. 2 (4) (2015) 285–302.

[10] M. Alweshah, Construction biogeography-based optimization algorithm for solving classification problems, Neural Comput. Appl. 31 (10) (2019) 5679–5688.

[11] M. Alweshah, A. Al-Daradkeh, M.A. Al-Betar, A. Almomani, S. Oqeili, $$\beta $$ β-Hill climbing algorithm with probabilistic neural network for classification problems, J. Ambient Intell. Humaniz. Comput. (2019) 1–12.

[12] M. Alweshah, Firefly algorithm with artificial neural network for time series problems, Res. J. Appl. Sci. Eng. Technol. 7 (19) (2014) 3978–3982.

[13] M. Alweshah, S. Abdullah, Hybridizing firefly algorithms with a probabilistic neural network for solving classification problems, Appl. Soft Comput. 35 (2015) 513–524.

[14] V.V. Kolisetty, D.S. Rajput, A review on the significance of machine learning for data analysis in big data, Jordan. J. Comput. Inf. Technol. (JJCIT) 6 (01) (2020).

[15] M. Seifzadeh, M. Salehi, B. Abedini, M.H. Ranjbar, The relationship between management characteristics and financial statement readability, EuroMed J. Bus. (2020).

[16] E.-S.M. El-Kenawy, A. Ibrahim, S. Mirjalili, M.M. Eid, S.E. Hussein, Novel feature selection and voting classifier algorithms for COVID-19 classification in CT images, IEEE Access 8 (2020) 179317–179335.

[17] E. Sivasankar, C. Selvi, S. Mahalakshmi, Rough set-based feature selection for credit risk prediction using weight-adjusted boosting ensemble method, Soft Comput. 24 (6) (2020) 3975–3988.

[18] W.M. Shaban, A.H. Rabie, A.I. Saleh, M. Abo-Elsoud, A new COVID-19 Patients Detection Strategy (CPDS) based on hybrid feature selection and enhanced KNN classifier, Knowl.-Based Syst. 205 (2020) 106270.

[19] A. Deniz, H.E. Kiziloz, On initial population generation in feature subset selection, Expert Syst. Appl. 137 (2019) 11–21.

[20] M. Alweshah, S. Alkhalaileh, D. Albashish, M. Mafarja, Q. Bsoul, O. Dorgham, A hybrid mine blast algorithm for feature selection problems, Soft Comput. (2020) 1–18.

[21] E.H. Houssein, M.E. Hosney, D. Oliva, W.M. Mohamed, M. Hassaballah, A novel hybrid Harris hawks optimization and support vector machines for drug design and discovery, Comput. Chem. Eng. 133 (2020) 106656.

[22] B. Mohammed, et al., Edge computing intelligence using robust feature selection for network traffic classification in Internet-of-Things, IEEE Access 8 (2020) 224059–224070.

[23] S. Karasu, A. Altan, S. Bekiros, W. Ahmad, A new forecasting model with wrapper-based feature selection approach using multi-objective optimization technique for chaotic crude oil time series, Energy 212 (2020) 118750.

[24] A. Kumar, et al., A novel health indicator developed using filter-based feature selection algorithm for the identification of rotor defects, Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability (2020) 1748006X20916953.

[25] Y. Fu, X. Liu, S. Sarkar, T. Wu, Gaussian mixture model with feature selection: An embedded approach, Comput. Ind. Eng. 152 (2020) 107000.

[26] F. Moslehi, A. Haeri, A novel hybrid wrapper–filter approach based on genetic algorithm, particle swarm optimization for feature subset selection, J. Ambient Intell. Humaniz. Comput. 11 (3) (2020) 1105–1127.

[27] M. Rostami, K. Berahmand, S. Forouzandeh, A novel community detection based genetic algorithm for feature selection, J. Big Data 8 (1) (2021) 1–27.

[28] !!! INVALID CITATION !!!.

[29] J. Li, H. Lei, A.H. Alavi, G.-G. Wang, Elephant herding optimization: variants, hybrids, and applications, Mathematics 8 (9) (2020) 1415.

[30] Y. Feng, S. Deb, G.-G. Wang, A.H. Alavi, Monarch butterfly optimization: A comprehensive review, Expert Syst. Appl. (2020) 114418.

[31] H. Al Nsour, M. Alweshah, A.I. Hammouri, H. Al Ofeishat, S. Mirjalili, A hybrid grey wolf optimiser algorithm for solving time series classification problems, J. Intell. Syst. 29 (1) (2018) 846–857.

[32] M. Alweshah, A. Omar, J. Alzubi, S. Alaqeel, Solving attribute reduction problem using wrapper genetic programming, Int. J. Comput. Sci. Netw. Secur. 16 (5) (2016) 77–84.

[33] M. Abdel-Basset, L. Abdel-Fatah, A.K. Sangaiah, Metaheuristic algorithms: A comprehensive review, in: Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications, Elsevier, 2018, pp. 185–231.

[34] S. Shilaskar, A. Ghatol, Feature selection for medical diagnosis: Evaluation for cardiovascular diseases, Expert Syst. Appl. 40 (10) (2013) 4146–4153.

[35] A.H. Halim, I. Ismail, S. Das, Performance assessment of the metaheuristic optimization algorithms: an exhaustive review, Artif. Intell. Rev. (2020) 1–87.

[36] T. Dokeroglu, E. Sevinc, T. Kucukyilmaz, A. Cosar, A survey on new generation metaheuristic algorithms, Comput. Ind. Eng. 137 (2019) 106040.

[37] G.-G. Wang, A.H. Gandomi, A.H. Alavi, D. Gong, A comprehensive review of krill herd algorithm: variants, Hybrids Appl. Artif. Intell. Rev. 51 (1) (2019) 119–148.

[38] B.H. Abed-alguni, Bat Q-learning algorithm, Jordan. J. Comput. Inf. Technol. (JJCIT) 3 (1) (2017) 56–77.

[39] A.H. Gandomi, X.-S. Yang, S. Talatahari, A.H. Alavi, Metaheuristic algorithms in modeling and optimization, Metaheuristic Appl. Struct. Infrastruct. (2013) 1–24.

[40] B.H. Abed-alguni, M. Barhoush, Distributed grey wolf optimizer for numerical optimization problems, Jordan. J. Comput. Inf. Technol.(JJCIT) 4 (2018) 130–149.

[41] N. Alshdaifat, M. Bataineh, Optimizing and thinning planar arrays using Chebyshev distribution and improved particle swarm optimization, Jordan. J. Comput. Inf. Technol. (JJCIT) 1 (1) (2015) 31–40.

[42] H.R. Lourenço, O.C. Martin, T. Stützle, Iterated local search, in: Handbook of Metaheuristics, Springer, 2003, pp. 320–353.

[43] M. Alweshah, Solving feature selection problems by combining mutation and crossover operations with the monarch butterfly optimization algorithm, Appl. Intell. (2020) 1–24.

[44] M. Alweshah, M. Al-Sendah, O.M. Dorgham, A. Al-Momani, S. Tedmori, Improved water cycle algorithm with probabilistic neural network to solve classification problems, Cluster Comput. 23 (4) (2020) 2703–2718.

[45] M. Alweshah, S.A. Khalaileh, B.B. Gupta, A. Almomani, A.I. Hammouri, M.A. Al-Betar, The monarch butterfly optimization algorithm for solving feature selection problems, Neural Comput. Appl. (2020).

[46] G.I. Sayed, A. Tharwat, A.E. Hassanien, Chaotic dragonfly algorithm: an improved metaheuristic algorithm for feature selection, Appl. Intell. 49 (1) (2019) 188–205.

[47] M.M. Mafarja, S. Mirjalili, Hybrid whale optimization algorithm with simulated annealing for feature selection, Neurocomputing 260 (2017) 302–312.

[48] V. Kumar, A. Kaur, Binary spotted hyena optimizer and its application to feature selection, J. Ambient Intell. Humaniz. Comput. 11 (7) (2020) 2625–2645.

[49] K.K. Ghosh, R. Guha, S. Ghosh, S.K. Bera, R. Sarkar, Atom Search Optimization with simulated Annealing–a Hybrid metaheuristic approach for feature selection, 2020, arXiv preprint arXiv:2005.08642.

[50] Y. Gao, Y. Zhou, Q. Luo, An efficient binary equilibrium optimizer algorithm for feature selection, IEEE Access 8 (2020) 140936–140963.

[51] S. Pichai, K. Sunat, S. Chiewchanwattana, An asymmetric Chaotic Competitive Swarm Optimization Algorithm for feature selection in high-dimensional data, Symmetry 12 (11) (2020) 1782.

[52] M. Sharma, P. Kaur, A comprehensive analysis of nature-inspired meta-heuristic techniques for feature selection problem, Arch. Comput. Methods Eng. (2020) 1–25.

[53] L. Brezočnik, I. Fister, V. Podgorelec, Swarm intelligence algorithms for feature selection: a review, Appl. Sci. 8 (9) (2018) 1521.

[54] S. Rajamohana, K. Umamaheswari, Hybrid approach of improved binary particle swarm optimization and shuffled frog leaping for feature selection, Comput. Electr. Eng. 67 (2018) 497–508.

[55] B. Venkatesh, J. Anuradha, A review of feature selection and its methods, Cybern. Inf. Technol. 19 (1) (2019) 3–26.

[56] S.P. Rajamohana, K. Umamaheswari, B. Abirami, Adaptive binary flower pollination algorithm for feature selection in review spam detection, in: 2017 International Conference on Innovations in Green Energy and Healthcare Technologies (IGEHT), IEEE, 2017, pp. 1–4.

[57] A. Zakeri, A. Hokmabadi, Efficient feature selection method using real-valued grasshopper optimization algorithm, Expert Syst. Appl. 119 (2019) 61–72.

[58] A. Bansal, A. Jain, Comparison of meta-heuristic with evolutionary and local search methods for feature selection, in: Metaheuristic and Evolutionary Computation: Algorithms and Applications, Springer, 2021, pp. 529–554.

[59] R. Millham, I.E. Agbehadji, H. Yang, Parameter tuning onto recurrent neural network and long short-term memory (RNN-LSTM) network for feature selection in classification of high-dimensional bioinformatics datasets, in: Bio-Inspired Algorithms for Data Streaming and Visualization, Big Data Management, and Fog Computing, Springer, 2021, pp. 21–42.

[60] M.A. Al-Betar, Z.A.A. Alyasseri, M.A. Awadallah, I.A. Doush, Coronavirus herd immunity optimizer (CHIO), 2020.

[61] A. Almomani, M. Alweshah, S. Al, Metaheuristic algorithms-based feature selection approach for intrusion detection, machine learning for computer and cyber security: principle, algorithms, and practices, 2019.

[62] S.M. Abubakar, Z. Sufyanu, M.M. Abubakar, A survey of feature selection methods for software defect prediction models, Fudma J. Sci. 4 (1) (2020) 62–68.

[63] M. Lui, T. Baldwin, Cross-domain feature selection for language identification, in: Proceedings of 5th international joint conference on natural language processing, 2011, pp. 553–561.
[64] H. Alazzam, A. Sharieh, K.E. Sabri, A feature selection algorithm for intrusion detection system based on pigeon inspired optimizer, Expert Syst. Appl. 148 (2020) 113249.
[65] A. Alarifi, A. Tolba, Z. Al-Makhadmeh, W. Said, A big data approach to sentiment analysis using greedy feature selection with cat swarm optimization-based long short-term memory neural networks, J. Supercomput. 76 (6) (2020) 4414–4429.
[66] M. Toğaçar, B. Ergen, Z. Cömert, F. Özyurt, A deep feature learning model for pneumonia detection applying a combination of mRMR feature selection and machine learning models, Irbm 41 (4) (2020) 212–222.
[67] S.R. Ahmad, A.A. Bakar, M.R. Yaakub, A review of feature selection techniques in sentiment analysis, Intell. Data Anal. 23 (1) (2019) 159–189.
[68] Q. Li, et al., An enhanced grey wolf optimization based feature selection wrapped kernel extreme learning machine for medical diagnosis, Comput. Math. Methods Med. 2017 (2017).
[69] Z. Zuo, J. Li, N.A. Moubayed, Curvature-based feature selection with application in classifying electronic health records, 2021, arXiv preprint arXiv:2101.03581.
[70] A.M. Anter, M. Ali, Feature selection strategy based on hybrid crow search optimization algorithm integrated with chaos theory and fuzzy c-means algorithm for medical diagnosis problems, Soft Comput. 24 (3) (2020) 1565–1584.
[71] M. Wang, H. Chen, Chaotic multi-swarm whale optimizer boosted support vector machine for medical diagnosis, Appl. Soft Comput. 88 (2020) 105946.
[72] M. Rostami, S. Forouzandeh, K. Berahmand, M. Soltani, Integration of multi-objective PSO based feature selection and node centrality for medical datasets, Genomics 112 (6) (2020) 4370–4384.
[73] A. Verma, M.K. Hanawal, N. Hemachandra, Unsupervised online feature selection for cost-sensitive medical diagnosis, in: 2020 International Conference on COMmunication Systems & NETworkS (COMSNETS), IEEE, 2020, pp. 1–6.
[74] A.K. Verma, S. Pal, S. Kumar, Prediction of skin disease using ensemble data mining techniques and feature selection method—a comparative study, Appl. Biochem. Biotechnol. 190 (2) (2020) 341–359.
[75] R. Kuppuchamy, M. Mangayarkarasi, A threshold fuzzy entropy based feature selection approach for breast cancer diagnosis.
[76] M.A. Rahman, R.C. Muniyandi, An enhancement in cancer classification accuracy using a two-step feature selection method based on artificial neural networks with 15 neurons, Symmetry 12 (2) (2020) 271.
[77] M.D. de Lima, J.d.O.R.e Lima, R.M. Barbosa, Medical data set classification using a new feature selection algorithm combined with twin-bounded support vector machine, Med. Biol. Eng. Comput. 58 (3) (2020) 519–528.
[78] J. Too, S. Mirjalili, A hyper learning binary dragonfly algorithm for feature selection: A COVID-19 case study, Knowl.-Based Syst. 212 (2020) 106553.
[79] R.A. Khurmaa, I. Aljarah, A. Sharieh, An intelligent feature selection approach based on moth flame optimization for medical diagnosis, Neural Comput. Appl. (2020) 1–40.
[80] Z. Yang, W. Wang, M. Shi, Algorithms and complexity for a class of combinatorial optimization problems with labelling, J. Optim. Theory Appl. 1–23.
[81] A.F. Henwood, Coronavirus disinfection in histopathology, J. Histotechnol. 43 (2) (2020) 102–104.
[82] T. Britton, F. Ball, P. Trapman, A mathematical model reveals the influence of population heterogeneity on herd immunity to SARS-CoV-2, Science 369 (6505) (2020) 846–849.
[83] D.R. Smith, Herd Immunity, the Veterinary clinics of North America, Food Anim. Pract. 35 (3) (2019) 593–604.
[84] M.G.M. Gomes, et al., Individual variation in susceptibility or exposure to SARS-CoV-2 lowers the herd immunity threshold, 2020, MedRxiv.
[85] V.J. Clemente-Suárez, et al., Dynamics of population immunity due to the herd Effect in the COVID-19 pandemic, Vaccines 8 (2) (2020) 236.
[86] J.S. Lavine, A.A. King, O.N. Bjørnstad, Natural immune boosting in pertussis dynamics and the potential for long-term vaccine failure, Proc. Natl. Acad. Sci. 108 (17) (2011) 7259–7264.
[87] O. Gokalp, E. Tasci, A. Ugur, A novel wrapper feature selection algorithm based on iterated greedy metaheuristic for sentiment classification, Expert Syst. Appl. 146 (2020) 113176.
[88] B. Koohestani, A crossover operator for improving the efficiency of permutation-based genetic algorithms, Expert Syst. Appl. 151 (2020) 113381.
[89] J. Thomas, N.S. Chaudhari, Selection of efficient crossover operator in metaheuristic approach for 2D strip packing, in: 2013 IEEE International Conference on Systems, Man, and Cybernetics, IEEE, 2013, pp. 415–420.
[90] Q. Wu, J.-K. Hao, A hybrid metaheuristic method for the maximum diversity problem, European J. Oper. Res. 231 (2) (2013) 452–464.
[91] F.B. Ozsoydan, M. Sağir, Iterated greedy algorithms enhanced by hyper-heuristic based learning for hybrid flexible flowshop scheduling problem with sequence dependent setup times: a case study at a manufacturing plant, Comput. Oper. Res. 125 (2021) 105044.
[92] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I.H. Witten, The WEKA data mining software: an update, ACM SIGKDD Explor. Newsl. 11 (1) (2009) 10–18.