Hindawi
Computational and Mathematical Methods in Medicine
Volume 2021, Article ID 6662420, 18 pages
https://doi.org/10.1155/2021/6662420

Research Article

Feature Selection and Classification of Clinical Datasets Using Bioinspired Algorithms and Super Learner

S. Murugesan,[1] R. S. Bhuvaneswaran,[1] H. Khanna Nehemiah,[1] S. Keerthana Sankari,[2] and Y. Nancy Jane[3]

[1] *Ramanujan Computing Centre, Anna University, Chennai 600025, India*
[2] *Department of Computer Science and Engineering, Anna University, Chennai 600025, India*
[3] *Department of Computer Technology, Anna University, Chennai 600025, India*

Correspondence should be addressed to S. Murugesan; bsmurugesan@gmail.com

Received 19 December 2020; Revised 10 April 2021; Accepted 23 April 2021; Published 18 May 2021

Academic Editor: David Diller

A computer-aided diagnosis (CAD) system that employs a super learner to diagnose the presence or absence of a disease has been developed. Each clinical dataset is preprocessed and split into training set (60%) and testing set (40%). A wrapper approach that uses three bioinspired algorithms, namely, cat swarm optimization (CSO), krill herd (KH) ,and bacterial foraging optimization (BFO) with the classification accuracy of support vector machine (SVM) as the fitness function has been used for feature selection. The selected features of each bioinspired algorithm are stored in three separate databases. The features selected by each bioinspired algorithm are used to train three back propagation neural networks (BPNN) independently using the conjugate gradient algorithm (CGA). Classifier testing is performed by using the testing set on each trained classifier, and the diagnostic results obtained are used to evaluate the performance of each classifier. The classification results obtained for each instance of the testing set of the three classifiers and the class label associated with each instance of the testing set will be the candidate instances for training and testing the super learner. The training set comprises of 80% of the instances, and the testing set comprises of 20% of the instances. Experimentation has been carried out using seven clinical datasets from the University of California Irvine (UCI) machine learning repository. The super learner has achieved a classification accuracy of 96.83% for Wisconsin diagnostic breast cancer dataset (WDBC), 86.36% for Statlog heart disease dataset (SHD), 94.74% for hepatocellular carcinoma dataset (HCC), 90.48% for hepatitis dataset (HD), 81.82% for vertebral column dataset (VCD), 84% for Cleveland heart disease dataset (CHD), and 70% for Indian liver patient dataset (ILP).

1. Introduction

Data related to symptoms observed on a patient at a point of time are stored in electronic health records (EHRs). Interesting patterns can be extracted from the data that are stored in EHRs, and the extracted patterns can be represented as knowledge, and this knowledge can assist the physicians to diagnose the presence or absence of a disease. Data mining tasks, namely, association rule mining, classification, and clustering are used to mine valuable patterns from the data stored in EHRs. Clinical decision support systems (CDSSs) that assist the physicians to diagnose the presence or absence of a disease can be developed from data stored in EHRs using bioinspired algorithms and data mining techniques. Although several algorithms have been proposed by researchers for association rule mining, classification, and clustering, no algorithm can be deliberated to be the "universal best." Quality of data and data distribution are the two key factors that determine the effectiveness of a data mining task. The performance of a data mining task depends on how effective data preprocessing has been done. Classification plays a major role in the development of CDSSs. Classification is a two-step process, first, building the classifier and second, model usage. Building the classifier is the process of training the classifier with a supervised learning algorithm. Model usage is the process of estimating the accuracy of the classifier using testing instances commonly referred to as testing set. Overfitting and underfitting are two major problems associated with building the classifier.

Clinical dataset (s) (C_s) used for classifier construction is split into a training set (T_s) and a testing set (T_t). Researchers have proposed different methods to identify the T_s and T_t. One common method is to split 80% of the dataset into T_s and 20% of the dataset into T_t. For clinical decision-making, a balanced dataset is essential for building a prediction model. Clinical datasets are normally not balanced, and classification methods perform poorly on minority class samples when the dataset is tremendously imbalanced. For example, consider a C_s with n instances, each instance associated with a class label c_1 or c_2. Among the n instances that 75% of the instances in C_s are associated with class label c_1, and 25% of the instances in C_s are associated with class label c_2, it is evident that the class labels in C_s are not equally represented and therefore, the C_s is imbalanced. In this context, c_1 is the majority class, and c_2 is the minority class, and hence, constructing a classifier with class-imbalanced data will lead to bias in favor of the majority class. One method to handle class imbalance in a C_s is to generate additional instances from the minority class. The Synthetic Minority Oversampling Technique (SMOTE) [1] is one of the prevailing methods used to generate additional training and testing instances.

A training instance can be defined as a tuple $t_i(f_1, f_2, \cdots f_m)$, where t_i represents a training instance, and $(f_1, f_2, \cdots f_j)$ represents the features corresponding to a training instance. The subscript i in t_i can range from 1 to n, where n is the number of instances. The subscript j in f_j can range from 1 to m, where m is the number of features. Using irrelevant features to train a classifier will affect its performance. Selecting the optimal features from the C_s and then training the classifier will enhance the accuracy of the classifier. Feature selection methods can be supervised, unsupervised, and semisupervised depending upon whether the training set is labeled or not. Commonly used supervised feature selection methods are filter and wrapper methods. The filter method considers the dependency of each feature to the class label and is independent of any classification algorithm. Measures, namely, information gain [2], gain ratio [3], Gini index [4], Laplacian score [5], and cosine similarity [6] can be used to rank the features. Other measures to rank the features can also be used in filter method. The wrapper method considers the classification accuracy of a learning algorithm to select the relevant features. Researchers are using a confluence of disciplines to develop computer-aided diagnostic (CAD) systems to assist physicians.

Knowledge mining using rough sets for feature selection and backpropagation neural network (BPNN) for classifying clinical datasets has been proposed in [7]. A CDSS to diagnose Urticaria using Bayes classification is proposed in [8]. CDSSs to diagnose lung disorders are proposed in [9–14]. A CDSS to diagnose the severity of gait disturbances using a Q-backpropogated time delay neural network on patients affected by Parkinson's disease is proposed in [15]. A statistical tolerance rough set induced decision tree classifier to classify multivariate time series clinical data is proposed in [16]. A CDSS to diagnose gestational diabetes mellitus using the fuzzy logic and radial basis function neural network is proposed in [17]. Use of fuzzy sets and extreme learning machine to classify clinical datasets is proposed in [18]. Wind-driven swarm optimization, a metaheuristic method to classify clinical datasets, is proposed in [19]. A computer-aided diagnostic system that uses a neural network classifier trained using differential evolution, particle swarm optimization, and gradient descent backpropagation algorithms is proposed in [20]. A radial basis function neural network to classify clinical datasets using k-means clustering algorithm and quantum-behaved particle swarm optimization is proposed in [21]. Classifying clinical unevenly spaced time series data by imputing missing values has been proposed in [22]. A framework to classify unevenly spaced time series clinical data using improved double exponential smoothing, rough sets, neural network, and fuzzy logic is proposed in [23].

An outline of nature-inspired algorithms for optimization is presented in [24]. The cooperative intellectual actions of insects or animal groups in nature, for example, colonies of ants, schools of fish, flock of birds, swarms of bees, and termites, have fascinated the thoughtfulness of researchers. Entomologists have studied the collective actions of insects or animals to model biological swarms, and engineers have applied these models as a framework to solve complex real-world problems.

In this work, a CAD system that employs a super learner to diagnose the presence or absence of a disease has been proposed. The bioinspired algorithms used in this work are cat swarm optimization (CSO), krill herd (KH), and bacterial foraging optimization (BFO). The classifiers used in this work are support vector machine (SVM) and BPNN trained using the conjugate gradient algorithm.

The rest of the paper is organized as follows: the abbreviation used in the manuscript is presented in Section 2. An outline of the related work is presented in Section 3. An outline of the datasets used is presented in Section 4. The framework of the proposed classifier is presented in Section 5. The results and discussions are presented in Section 6. Finally, conclusion and scope for future work are presented in section 7.

2. Abbreviations Used

Table 1 presents the abbreviation used in the rest of the manuscript in alphabetic order.

3. Literature Survey

Leema et al. [25] in their work have experimented the significance of fixing the appropriate values of parameters to train artificial neural networks using the backpropagation algorithm. The parameters are initial weight selection, bias, activation function used, number of hidden layers, number of neurons per hidden layer, number of training epochs, minimum error, and momentum term. Twelve backpropagation learning algorithms have been used in this study. Experimentation has been carried out using three clinical

Table 1: Abbreviations used.

Abbreviation	Phrase
ABCO	Artificial bee colony optimization
ACO	Ant colony optimization
ANN	Artificial neural networks
BCS	Binary cuckoo search
BFA	Binary firefly algorithm
BFO	Bacterial foraging optimization
BP	Back propagation
BPNN	Back propagation neural network
CAD	Computer-aided diagnosis
CDC	Counts of dimension to change
CDSS	Clinical decision support system
CFCSA	Hybrid crow search optimization algorithm
CGA	Conjugate gradient algorithm
CHD	Cleveland heart disease
CMVO	Chaotic multiverse optimization
CSM	Cosine similarity measure
CSO	Cat swarm optimization
CT	Computed tomography
DE	Differential evolution
DGA	Distance-based genetic algorithm
DISON	Diverse intensified strawberry optimized neural network
DNN	Deep neural network
E.coli	Escherichia Coli Bacteria
ECSA	Enhanced crow search algorithm
ELM	Extreme learning machine
FBFO	Feature selected by bacterial foraging optimization
FCM	Fuzzy C-means
FCSO	Feature selected by cat swarm optimization
FFO	Firefly optimization
FKH	Feature selected by krill herd
GA	Genetic algorithm
GSO	Glowworm swarm optimization
HCC	Hepatocellular carcinoma
HD	Hepatitis
IBPSO	Improved binary particle swarm optimization
ILP	Indian liver patient
ISSA	Improved Salp swarm algorithm
KH	Krill herd
k-NN	k-nearest neighbors
LO	Lion optimization
LR	Logistic regression
MCC	Mathew's correlation coefficient
MFO	Moth-flame optimization
ML	Machine learning
MPNN	Multilayer perceptron neural network
MR	Mixed ratio
NB	Naive Bayes
PCC	Pearson correlation coefficient
PID	Pima Indian diabetes
PSO	Particle swarm optimization
RD	Random diffusion
RDM	Rough dependency measure
RF	Random forest
RoIs	Regions of interest
SHD	Statlog heart disease
SMOTE	Synthetic minority oversampling technique
SMP	Seeking memory pool
SPC	Self-position consideration
SRD	Seeking range of the selected dimension
SVC	Support vector classification
SVM	Support vector machine
TS	Thoracic surgery
UCI	University of California Irvine
VCD	Vertebral column dataset
WBC	Wisconsin breast cancer
WDBC	Wisconsin diagnostic breast cancer
WOA	Whale optimization algorithm

datasets from the UCI ML repository, namely, PID, hepatitis, and WBC datasets.

Elgin et al. [26] in their work have proposed a clinical-decision making system to diagnose allergic rhinitis. A wrapper approach that uses GA and the accuracy of ELM classifier as the fitness function has been used for feature selection. The selected features have been trained using ELM classifier. Intradermal skin test dataset of 872 patients collected from Good Samaritan Lab Services and Allergy Testing Centre, Chennai, has been used in this work, and an accuracy of 97.7% has been achieved.

Sreejith et al. [27] in their work have proposed a framework for classifying clinical datasets which uses an embedded approach for feature selection and a DISON for classification. The feature selection is performed by computing the feature importance of every attribute using an extremely randomized tree classifier. Classification is performed using DISON which is a feed forward neural network whose weights and bias are optimized in two stages first, by using a strawberry optimization algorithm and then by using a gradient descent BP algorithm. Vertebral column, PID, CHD, and SHD datasets from the UCI ML repository have been used for experimentation. The framework has achieved an accuracy of 87.17% for vertebral column, 90.92% for PID, 93.67% for CHD, and 94.5% for SHD.

Sreejith et al. [28] in their work have proposed a framework for CDSS which addresses the data imbalance problems associated with clinical dataset. The datasets are rebalanced using SMOTE enhanced using Orchard's algorithm. The feature selection is performed using a wrapper approach where CMVO is used to select the feature subsets, and RF classifier is used to evaluate the goodness of the features. The arithmetic mean of MCC and F-score computed using the RF

Table 2: Outline of the datasets used.

Dataset name	No. of instances	No. of features*	No. of missing values	Class labels with no. of instances associated with each class label	Interpretation of class labels
WDBC	569	31	Nil	M (212)/B (357)	M-malignant, B-benign
SHD	270	13	Nil	2 (120)/1 (150)	2-present, 1-absent
HCC	165	49	826	0 (63)/1 (102)	0-dies, 1-lives
HD	155	18	167	1 (32)/2 (123)	1-die, 2-live
VCD	310	6	Nil	0 (210)/1 (100)	0-abnormal, 1-normal
CHD	303	13	Nil	1 (139)/2 (164)	1-presence, 2-absence
ILP	583	10	Nil	1 (416)/2 (167)	1-diseased, 2-nondiseased

* without class label.

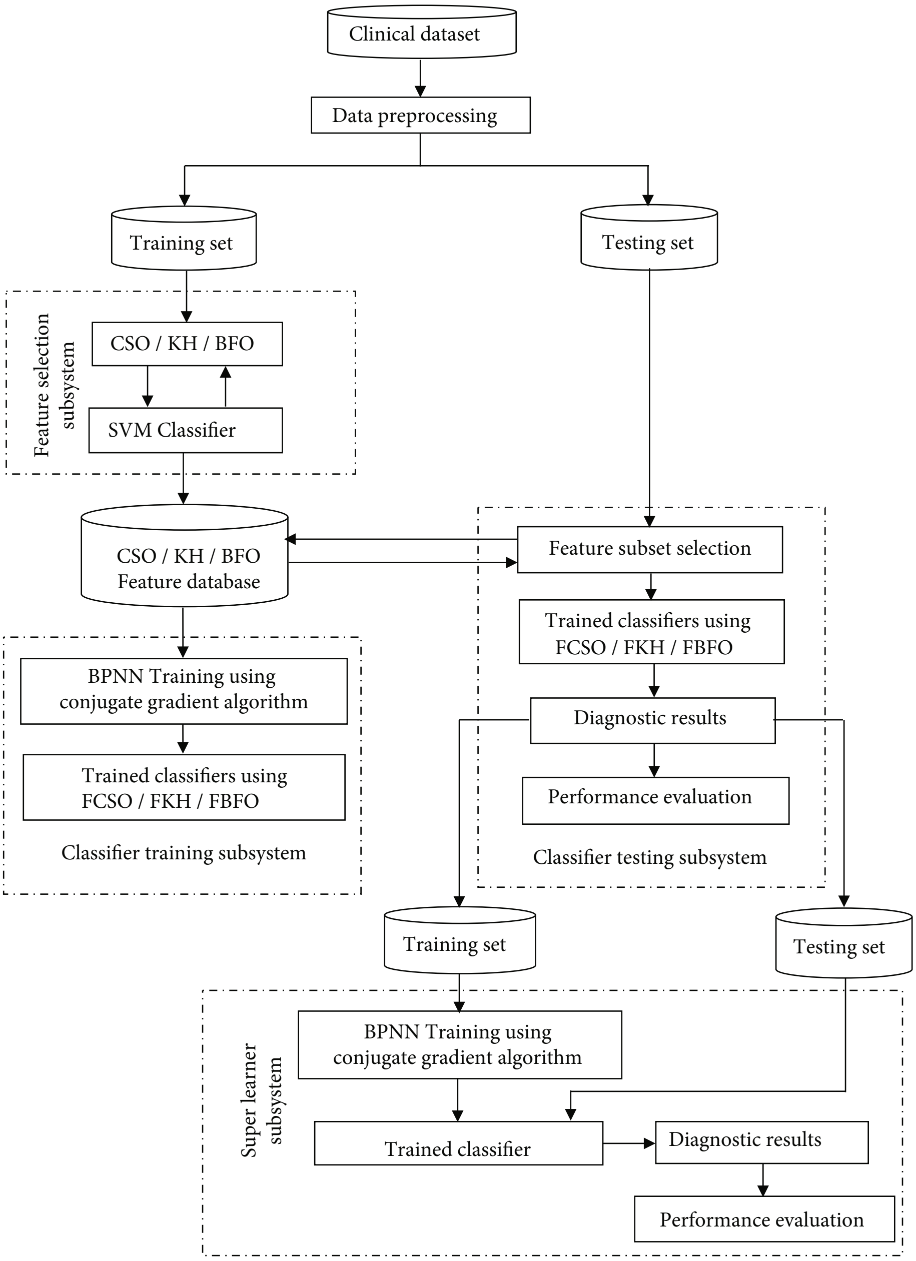
Clinical dataset
Data preprocessing
Training set
Testing set
Feature selection subsystem
CSO / KH / BFO
SVM Classifier
CSO / KH / BFO Feature database
Feature subset selection
Trained classifiers using FCSO / FKH / FBFO
Diagnostic results
Performance evaluation
Classifier testing subsystem
BPNN Training using conjugate gradient algorithm
Trained classifiers using FCSO / FKH / FBFO
Classifier training subsystem
Training set
Testing set
Super learner subsystem
BPNN Training using conjugate gradient algorithm
Trained classifier
Diagnostic results
Performance evaluation

Figure 1: System framework.

Table 3: Outline of training and testing instances of each C_s.

Instances	WDBC dataset	SHD dataset	HCC dataset	HD dataset	VCD dataset	CHD dataset	ILP dataset
Total number of instances before SMOTE	569	270	165	155	310	303	583
Total number of instances after SMOTE	780	270	228	251	410	303	750
Number of training instances for FCSO/FKH/FBFO classifiers	468	162	137	151	246	182	450
	60% of the total number of instances after SMOTE						
Number of testing instances for FCSO/FKH/FBFO classifiers	312	108	91	100	164	121	300
	40% of the total number of instances after SMOTE						
Number of training instances for super learner	250	86	73	80	131	97	240
	80% of the total testing instances* for FCSO/FKH/FBFO classifiers						
Number of testing instances for Super learner	62	22	18	20	33	24	60
	20% of the total testing instances* for FCSO/FKH/FBFO classifiers						

*Each instance refers to the classification result pertaining to each instance of the testing set for FCSO, FKH, and FBFO classifiers and the class label corresponding to each instance of the testing set.

Table 4: Outline of the parameters used in CSO.

Parameter	Description
SMP	SMP is used to define the size of the seeking memory of each cat. Each cat selects possible neighborhood position from a set of solutions.
SRD	SRD is used to define the seeking range of the selected dimension.
CDC	CDC is a count of dimensions to be changed in seeking mode.
SPC	SPC indicates whether the cat is in the current position or not.
N	Number of cats
MR	Mixed ratio of cats
C	Constant value
D	Size of dimension
R	Random number in the range of [0,1]

classifier is used as the fitness function. Finally, an RF classifier, comprising of 100 decision trees which uses information gain ratio as the split criteria, is used for classifying the clinical data. Three clinical datasets from the UCI ML repository, namely, ILP, TS, and PID datasets, have been used for experimentation. The proposed framework achieved 0.65 MCC, 0.84 *F*-score, and 82.46% accuracy for ILP; 0.74 MCC, 0.87 *F*-score, and 86.88% accuracy for TS; and 0.78 MCC, 0.89 *F*-score, and 89.04% accuracy for PID datasets.

Isaac et al. [29] in their work have proposed a CAD system to diagnose pulmonary emphysema from chest CT slices. Spatial intuitionistic fuzzy *C*-means clustering algorithm has been used to segment the lung parenchyma and extracting the RoIs. From the RoIs, shape, texture, and run-length features have been extracted, and feature selection has been performed using a wrapper approach using four bioinspired algorithms with the classification accuracy of SVM as the fitness function. The bioinspired algorithms used are MFO, FFO, ABCO, and ACO. Tenfold crossvalidation technique has been used, and each feature set has been trained using an ELM classifier. Two independent datasets, one dataset consisting of CT slices collected from hospitals and the second dataset consisting of CT slices from a benchmark repository, have been used for classification. A maximum classification accuracy of 89.19% for MFO, 91.89% for FFO, 83.78% for ABCO, 86.49% for ACO, and 75.68% without feature selection have been achieved.

Elgin et al. [30] in their work have performed feature selection and instance selection using a wrapper approach that employs cooperative coevolution with the classification accuracy of the random forest classifier as the fitness function. The optimal feature set is used to train a random forest classifier. Seven datasets, namely, WDBC, HD, PID, CHD, SHD, VCD, and HCC from the UCI ML repository have been used for experimentation. An accuracy of 97.1%, 82.3%, 81.01%, 93.4%, 96.8%, 91.4%, and 72.2% for datasets WDBC, HD, PID, CHD, SHD, VCD, and HCC datasets have been achieved, respectively.

Anter et al. [31] in their work have developed CFCSA by integrating chaos theory and the FCM method to find the optimal feature subset. Ten clinical datasets from the UCI ML repository have been used for experimentation. The features of each clinical dataset have been normalized, and then random chaotic motion has been incorporated into CFCSA in the form of chaotic maps. The objective function of the FCM has been used as the fitness function, in which the crow with the best fitness has been considered the best solution. Comparison has been done with chaotic ant lion optimization, binary ant lion optimization, and the binary crow search algorithm, and it has been inferred that CFCSA outperforms these algorithms in all the datasets used for experimentation.

Elgin et al. [32] in their work have proposed a correlation-based ensemble feature selection using a wrapper approach that employs three bioinspired algorithms using differential evolution, lion optimization, and glowworm swarm optimization with the accuracy of the AdaboostSVM classifier as the fitness function. Tenfold crossvalidation technique has been used, and the optimal features selected have been used to train a gradient descent BP neural network with variable learning rates. Two clinical datasets from the UCI

```
Input: training set
Process:
Step 1: initialize the population of $N$ cats (solutions) at random. Each solution is of length $n$, where $n$ represents the number of features. If the corresponding feature is selected, it is represented as "1;" else, it is represented as "0." Initialize the parameters, namely, SMP, SRD, CDC, SPC, MR, C, and R.
Step 2: calculate the fitness value of each cat (solution) using the SVM classifier, where the accuracy of the SVM classifier is considered as the fitness function. The solution that has the maximum fitness value obtained so far is considered as the best solution.
Step 3: assign the cats to perform seeking mode. Seeking mode refers to the cats at rest and its movement to the next position by looking around itself.
Step 3a: create $j$ (SMP) copies of the current cat. All the copies are considered to be candidate solutions.
Step 3a. i: if the value of SPC is true, one among the candidates retain the position, while the rest changes its position with respect to a randomly selected SRD.
Step 3a. ii: if the value of SPC is false, then all the candidates change their position by a randomly selected CDC.
Step 3b: calculate the probability of each solution being selected using Equation (2) to find the best solution that has the maximum chance to survive. If all the solutions produce the same fitness value, then the probability value is considered as "1."
$\mathrm{P}_\mathrm{i} = |\mathrm{FS}_\mathrm{i} - \mathrm{FS}_\mathrm{b}|/\mathrm{FS}_{\max} - \mathrm{FS}_{\min}$, where $0 < i < j$.(2)
In the above formula, $P_i$ is the probability of the current cat $i$, $FS_{\max}$ is the maximum fitness value, and $FS_{\min}$ is the minimum fitness value. The values of $\mathrm{FS}_b$ are assigned $\mathrm{FS}_{\max}(\mathrm{FS}_b = \mathrm{FS}_{\max})$ if maximum fitness has to be calculated. The values of $\mathrm{FS}_b$ are assigned $\mathrm{FS}_{\min}$ $(\mathrm{FS}_b = \mathrm{FS}_{\min})$ if minimum fitness has to be calculated. In our work, the value of $\mathrm{FS}_b$ is assigned to $\mathrm{FS}_{\max}$.
Step 4: perform tracing mode. In this mode, the cats update their position based on the velocity. Calculate the velocity $(V_{k,d}^{t+1})$ and update the position $(x_{k,d}^{t+1})$ of each cat using Equation (3) and Equation (4).
$V_{k,d}^{t+1} = V_{k,d}^{t} + R \times C \times (x_{\mathrm{best},d}^{t} - x_{k,d}^{t})$, °Where $d = 1, 2, \cdots.., D$,(3)
$x_{k,d}^{t+1} = x_{k,d}^{t} + V_{k,d}^{t}$.(4)
In the above formula, $x_{k,d}^{t}$ and $V_{k,d}^{t}$ are the position and velocities of current cat $k$ at iteration $t$. The best solution set from the cats in the population is denoted by $x_{\mathrm{best},d}^{t}$; $d$ denotes the dimension to be changed; $C$ is a constant, and $R$ is a random number between 0 and 1.
Step 5: update the best solution that has the maximum fitness value. If the solution in the previous iteration has low fitness value, then replace it with the current best solution; otherwise, retain the previous best solution.
Step 6: repeat step 2 to step 5 for a maximum number of iterations or until the convergence of solution is reached. The solution with the maximum fitness value obtained by the classifier is considered as the optimal feature subset.
Output: optimal feature subset.
```

Algorithm 1:

Table 5: Outline of the parameters used in the KH algorithm.

Parameter	Definition	Value
V_f	Maximum foraging speed	$V_f = 0.02\ \mathrm{m/s^{-1}}$
$\mathrm{RD}^{\max}$	Maximum random diffusion speed	$\mathrm{RD}^{\max} \in (0.002 - 0.01\ )\ \mathrm{m/s^{-1}}$
$N^{\max}$	Maximum induction speed	$N^{\max} = 0.01\ \mathrm{m/s^{-1}}$
w_n	Inertia weight of the motion induced	$w_n \in (0, 1)$
w_f	Inertia weight of the foraging motion	$w_f \in (0, 1)$
C_t	Step-length scaling factor	Constant no.between $[0, 2]$
δ	Random directional vector	Random numbers $[-1, 1]$

ML repository, namely, hepatitis and WDBC have been used for experimentation. An accuracy of 93.902% for hepatitis and 98.734% for WDBC datasets have been achieved.

Sweetlin et al. [33] in their work have proposed a CAD system to diagnose pulmonary tuberculosis from chest CT slices. The region growing algorithm has been used for segmenting the lung fields followed by edge reconstruction. The manifestations of pulmonary tuberculosis, namely, cavities, consolidations, and nodules have been considered to be RoIs. After extracting the RoIs, and from the RoI, texture features, run-length features and shape features have been extracted, and feature selection has been performed using a wrapper approach that employs the BCS algorithm with the accuracy of one-against-all multiclass SVM classifier as the fitness function. The Cuckoo search algorithm has been implemented in two ways, first, by using entropy measure and second, without using entropy measure. Using the selected feature training is performed using one-against-all multiclass SVM classifier. An accuracy of 85.54% for BCS algorithm with entropy measure and 84.65% accuracy for BCS algorithm without entropy measure have been achieved.

Sweetlin et al. [34] in their work have proposed a CAD system to diagnose pulmonary hamartoma nodules from chest CT slices. Otsu's thresholding method has been used

```
Input: training set
Process:
Step 1: initialize the population of $N$ krill herds (solutions) at random. Each solution is of length $n$, where $n$ represents the number of features. If the corresponding feature is selected, it is represented as "1," else as "0." Initialize the parameters maximum induced motion $N^{\max}$, foraging speed $V_f$, maximum random diffusion speed $RD^{\max}$, $w_n$, $w_f$, $C_t$, and $\delta$.
Step 2: calculate the fitness value of each krill herd (solution) using the SVM classifier, where the accuracy of the SVM classifier is considered as the fitness function. The solution with the highest fitness value is considered as the global best solution.
Step 3: update the position of each krill $(x_i\ (t + \Delta t))$ using Equations (6) and (7) based on movement induced by other krill individuals, foraging activity, and random diffusion.
$x_i\ (t + \Delta t) = x_i\ (t) + \Delta t(N_i + F_i + RD_i)$,(6)
Where$\Delta t = C_t \sum_{j=1}^{d} (UB_j - LB_j)$, where $j = 1, 2, \cdots .d$.(7)
In the above formula, $x_i(t)$ is the current position of the krill; $\Delta t$ is the scaling factor of the velocity vector; $N_i$ is the induced motion; $F_i$ is the foraging motion; RD$_i$ is the random diffusion of the $i^{th}$ krill individual; $C_t$ is the step-length scaling factor; $d$ is the total number of krill individuals; $UB_j$ is the upper bounds of variable $j$, and LB$_j$ is the lower bounds of variable $j$.
Step 4: each krill individual maintains a high density and change their position due to their mutual effect. The direction of individual krill is maintained by target effect, local effect, and repulsive effect. The induced movement $N_i$ by other krills is calculated using Equations (8) and (9).
$N_i = N^{\max} \times \alpha_i + w_n \times N_i^t$,(8)
Where, $\alpha_i = \alpha_i^{\text{local}} + \alpha_i^{\text{target}}$.(9)
In the above formula, $N_i$ is the induced motion; $N^{\max}$ is the maximum induction speed; $\alpha_i$ is the induced direction; $w_n$ is the inertia weight of the motion induced; $N_i^t$ is the last induced motion; $\alpha_i^{\text{local}}$ is estimated from the local effect, and $\alpha_i^{\text{target}}$ is the target effect.
Step 5: calculate the foraging motion $F_i$ using Equations (10) and (11). It is mainly based on the current location of the food and the previous experience about the food location.
$F_i = V_f \times \beta_i + w_f \times F_i^t$,(10)
Where, $\beta_i = \beta_i^{\text{food}} + \beta_i^{\text{best}}$.(11)
In the above formula, $V_f$ is the maximum foraging speed; $\beta_i$ is the foraging motion; $w_f$ is the inertia weight of the foraging motion; $F_i^t$ is the last foraging motion; $\beta_i^{food}$ is the food attractive, and $\beta_i^{\text{best}}$ is the effect of the best fitness of the $i^{th}$ krill.
Step 6: calculate the random motion for random diffusion $RD_i$ using Equation (12) which is characterized with high diffusion speed and a random vector.
$RD_i = RD_{\max} \times (1 - I/\ I_{\max}) \times \delta$.(12)
In the above formula, RD$^{\max}$ is the maximum random diffusion speed; $\delta$ is the random directional vector; $I$ is the current iteration number, and $I_{\max}$ is the maximum number of iterations.
Step 7: repeat steps 2 to 6 for a maximum number of iterations or until the convergence of solution is reached. The solution with the maximum fitness value obtained by the classifier is considered as the optimal feature subset.
Output: optimal feature subset.
```

Algorithm 2:

to segment lung parenchyma from the CT slices. Nodules are considered to be the RoIs and from the RoIs, texture features, shape features and run-length features have been extracted. Feature selection has been performed using filter evaluation measures, namely, CSM and RDM with the ACO algorithm. The features selected by ACO-CSM and ACO-RDM have been used to train three classifiers, namely, SVM, NB, and J48 decision tree classifiers. Maximum classification accuracy of 94.36% for SVM classifier trained with 38 features selected using ACO-RDM has been achieved.

Sweetlin et al. [35] in their work have proposed a CAD system to diagnose pulmonary bronchitis from CT slices of the lung. Optimal thresholding has been used to segment the left and right lung fields from the lung CT slices. The RoIs are identified, and from the RoIs, texture and shape features have been extracted. Feature selection has been performed using a hybrid ACO algorithm combined with tandem run recruitment based on cosine similarity, and the accuracy of the SVM classifier has been used as the fitness function. The selected features have been used to train a SVM classifier. An accuracy of 81.66% for ACO with tandem run strategy, 78.10% for ACO without tandem run strategy, and 75.14% without feature selection has been achieved.

Raj et al. [36] in their work have proposed DGA for feature selection to develop a CAD system to diagnose lung disorders from chest CT slices. The entire dataset has been split into two sets one set containing 90% of the entire dataset and the other set containing 10% of the entire dataset. Out of the 90%, 50% has been used as training set and the other 50% as validation set for evaluating the objective function. The set containing 10% of the entire dataset has been used as testing set. The objective function has been defined as the sum of the squared deviation of each data in the training set of each class from each data in the validation set of the corresponding class. GA has been used for feature selection by minimizing the proposed objective function, resulting in the proposed DGA. The GA has been iterated over several generations to obtain individuals that are best fit with respect to the objective function. Classification has been performed using k-NN classifier to classify the RoIs into one of four classes, namely, bronchiectasis,

Table 6: Outline of the parameters used in the BFO algorithm.

Parameter	Description
p	Number of features
S	Number of bacteria
S_r	Number of bacteria in the reproduction steps
N_{re}	No. of reproductive steps
N_{ed}	No. of elimination-dispersal steps
N_c	No. of chemotactic steps
N_s	No. of swimming steps
$L_{(i)}$	Bacteria step size length
P_{ed}	Elimination probability
$\varnothing(i)$	Direction of i^{th} bacteria
x	Index of the chemotactic process
y	Index of the reproduction process.
z	Index of the elimination-dispersal process
θ^i	The i^{th} bacterium position
θ	A bacterium on the optimization domain
J_{last}	The highest objective function value
$\Delta(i)$	A random vector and its value lie between -1 and 1
$J\text{cc}\left(\theta, \theta^i(x, y, z)\right)$	Cell-to-cell attractant effect to nutrient concentration

tuberculosis, pneumonia, and normal. An average accuracy of 88.16% with feature selection and an average accuracy of 86.46% without feature selection have been achieved.

Zawbaa et al. [37] in their work have performed feature selection using a wrapper approach that uses the MFO algorithm with the accuracy of k-NN classifier as the fitness function. Eighteen datasets from the UCI ML repository have been used for experimentation among which four are clinical datasets. Comparison has been done with PSO and GA, and it has been inferred that MFO outperforms in fourteen datasets among which three are clinical datasets.

Shu-Chuan et al. [38] in their work have presented an algorithm called CSO by modeling the natural behavior of cats. The CSO algorithm considered two biological characteristics of cats, namely, seeking mode and tracking mode. Cats spend utmost of the time when they are awake on resting. Nevertheless, during their rests, their perception is really high, and they are well aware of what is happening around them. Cats continuously observe their environment wisely and consciously and when they perceive a prey, they advance towards it rapidly. Although resting, they move their position cautiously and slowly, occasionally even stay in the original position. Seeking mode has been used to represent this behavior into the CSO, and the tracing mode has been used to represent the behavior of cats advancing towards a prey into the CSO. The performance of CSO has been evaluated by applying CSO, standard PSO, and PSO with weighting factor into six benchmark functions. The results obtained reveal that the proposed CSO performs better compared to PSO and PSO with weighting factor.

Gandomi et al. [39] in their work have proposed a swarm intelligence algorithm named KH algorithm to solve optimization tasks and is centered on the imitation of the herding behavior of krill swarms with respect to precise biological and environmental processes. The fitness function of each krill individual has been defined as the least distance of each individual krill from food and from the highest density of the herd. Three vital actions considered to define the time-dependent position of an individual krill are, one, movement induced by other krill individuals, two, foraging activity, and three, random diffusion. The KH algorithm is tested using twenty benchmark functions and compared with eight algorithms. Experimentation results indicate that the KH algorithm can outperform these familiar algorithms.

Chen et al. [40] have proposed a cooperative bacterial foraging optimization algorithm (CBFO). Two cooperative methods are used to solve complex optimization problems in the original BFO [41] and achieved significant improvement. The serial heterogeneous cooperation on the implicit space decomposition level and the hybrid space decomposition level are the two methods used to improve the original BFO. The authors have compared the performance of two CBFO variants with the original BFO, PSO, and GA on four commonly used benchmark functions. The experimental results indicated that the CBFO achieved a better performance over the original BFO, PSO, and GA.

Chen et al. [42] have proposed an adaptive bacterial foraging optimization (ABFO) for optimizing functions. The adaptive foraging approaches are used to increase the performance of the original BFO. It is achieved by enabling the original BFO to adjust the run-length unit parameter dynamically during the time of algorithm implementation. The experimental results are compared with the original BFO, PSO, and GA using 4 benchmark functions. The proposed ABFO indicates the better performance over the original BFO and competitive with the PSO and GA.

From the literature, it is evident that classifier training using relevant features enhances the accuracy of the classifier. It can also be inferred that wrapper-based feature selection that employs bioinspired algorithms performs better in numerous cases compared to traditional feature selection methods.

4. Outline of the Datasets Used

Seven clinical datasets from the UCI ML repository, namely, WDBC, SHD, HCC, HD, VCD, CHD, and ILP have been used for binary classification. An outline of each dataset used is presented in Table 2.

5. System Framework

The framework for feature selection and classification of clinical datasets using bioinspired algorithms and super learner is presented in Figure 1. The major building blocks of the framework are data preprocessing, feature selection, classifier training, classifier testing, and dataset construction for super learner, super learner training, and testing. Each building block is outlined below.

```
Input: training set
Process:
Step 1: initialize the population of $S$ bacteria (solutions) at random. Each solution is of length $p$, where $p$ represents the number of features. If the corresponding feature is selected, it is represented as "1," else as "0." Initialize the parameters $p, N_c, N_s, N_{re}, N_{ed}, P_{ed}$, and $L_{(i)}$ (where subscript $i$ in $L_{(i)}$ can range from 1,2,…S), $\theta^i$,$x = 0, y = 0$, and $z = 0$.
Step 2: calculate the fitness value of each bacterium (solution) using the SVM classifier, where the accuracy of the SVM classifier is considered as the fitness function.
Step 3: in the elimination-dispersal process, due to environmental changes, the bacteria are eliminated or dispersed from current location. This process is used to strengthen the ability of global optimization. Initiate the elimination-dispersal process and increase the value of $z$ from 0 to $N_{ed}$.
Step 4: in the reproduction process, the low healthy bacteria die and rest of the other healthiest bacteria are divided into two bacteria. The new bacteria are placed on the same position of their parent. this process is used to maintain the population rate of bacteria. Initiate the reproduction process and increase the value of $y$ from 0 to $N_{re}$.
Step 5: in the chemotactic process, the E. coli bacterium performs two actions during the entire life time, namely, tumble and swim. Initiate the chemotactic process and increase the value of $x$ from 0 to $N_c$.
Step 6: execute the chemotactic process. Each bacterium $i$ moves into a chemotactic process, where $i = 1, 2, \cdots .S$,
Step 6a: calculate the objective function $J(i, x, y, z)$ for each bacterium using Equation (13).
$J(i, x, y, z) = J(i, x, y, z) + Jcc(\theta, \theta^i(x, y, z)).(13)$
Assign the value of objective function $J(i, x, y, z)$ in to $J_{\text{last}}$.
Step 6b: perform tumbling action for each bacterium using Equation (14). This action will enable the bacteria to change the present direction for a period of time.
$\varnothing(i) = \Delta(i)/\sqrt{\Delta(i)^T\Delta(i)}.(14)$
In the above formula, $T$ is the maximum number iterations, and $\Delta(i)/\sqrt{\Delta(i)^T\Delta(i)}$ is a random forward direction of movement.
Step 6c: based on the tumbled direction obtained by the bacteria, each bacteria move to a random position using Equation (15).
$\theta^i(x + 1, y, z) = \theta^i(x, y, z) + L(i) \times \varnothing(i).(15)$
In the above formula, $\theta^i(x, y, z)$ is the $i^{th}$ bacterium position in the $x^{th}$ chemotaxis, $y^{th}$ reproduction, and $z^{th}$ elimination-dispersal procedure, and $\theta^i(x + 1, y, z)$ means the $i^{th}$ bacterium position in the $x + 1^{th}$ chemotaxis, $y^{th}$ reproduction, and $z^{th}$ elimination-dispersal procedure.
Step 6d: compute the objective function value $J(i, x + 1, y, z)$ using Equation (16).
$J(i, x + 1, y, z) = J(i, x + 1, y, z) + Jcc(\theta, \theta^i(x + 1, y, z)).(16)$
Step 6e: perform swim action and assign the value of swim length $m = 0$.
Step 6f: when the number of steps $Ns$ in the swim process is greater than the swim length (m), then increase the $m$ value to 1 $(m = m + 1)$. If the value of $J(i, x + 1, y, z) > J_{\text{last}}$ replace the $J_{\text{last}}$ value using the current best objective value $J(i, x + 1, y, z)$, then assign the swim length $m = Ns$.
Step 7: if $x < N_c$, then go to step 5. Else, go to step 8.
Step 8: execute the reproduction process. In this reproduction process, the accumulated cost of bacterium ( $J^i_{\text{health}}$ ) is calculated using equation (17).
$J^i_{\text{health}} = \sum_{x=1}^{N_c+1} J(i, x, y, z).(17)$
The accumulated cost of bacterium ( $J^i_{\text{health}}$ ) represents the health of the bacterium. Bacteria will be sorted in descending order based on the $J^i_{\text{health}}$ value. If the accumulated cost of bacterium is high, it means that the bacterium did not get enough nutrition or food during its entire lifetime. They are considered to have low health and set to die. The remaining healthy bacteria are divided into two. The reproduced bacteria are positioned at the same place as their parents.
Step 8a: if the number of defined reproduction steps $N_{re}$ is not achieved ( $y < N_{re}$ ), then go to step 4.
Step 9: execute the elimination-dispersal process. Based on the elimination probability ($P_{ed}$), this process is used to keep the number of bacteria in the population unchanged. If a bacterium is eliminated, a random search is initialized to move to a new position to avoid local optimum, after a certain number of reproduction movements.
Step 10: repeat the process from step 3 to step 9 until the number of elimination dispersal steps $(N_{ed})$ is greater than the value of $z$. Otherwise, terminate the process.
Output: optimal feature subset.
```

Algorithm 3:

5.1. Preprocessing. Each (C_s) has been subjected to preprocessing prior to feature selection to enhance the quality of data. Mean imputation has been used to handle missing values, and SMOTE is used to handle the class imbalance problem in each C_s by generating additional instances from the minority class.

Normalization has been used to scale the value of a feature so that the value will fall in a specified range and is predominantly useful for constructing a classifier involving a neural network. Training a classifier using normalized data will speedup learning. In this work, the range is 0 to 1, and min-max normalization is being used. When an attribute "A" in a

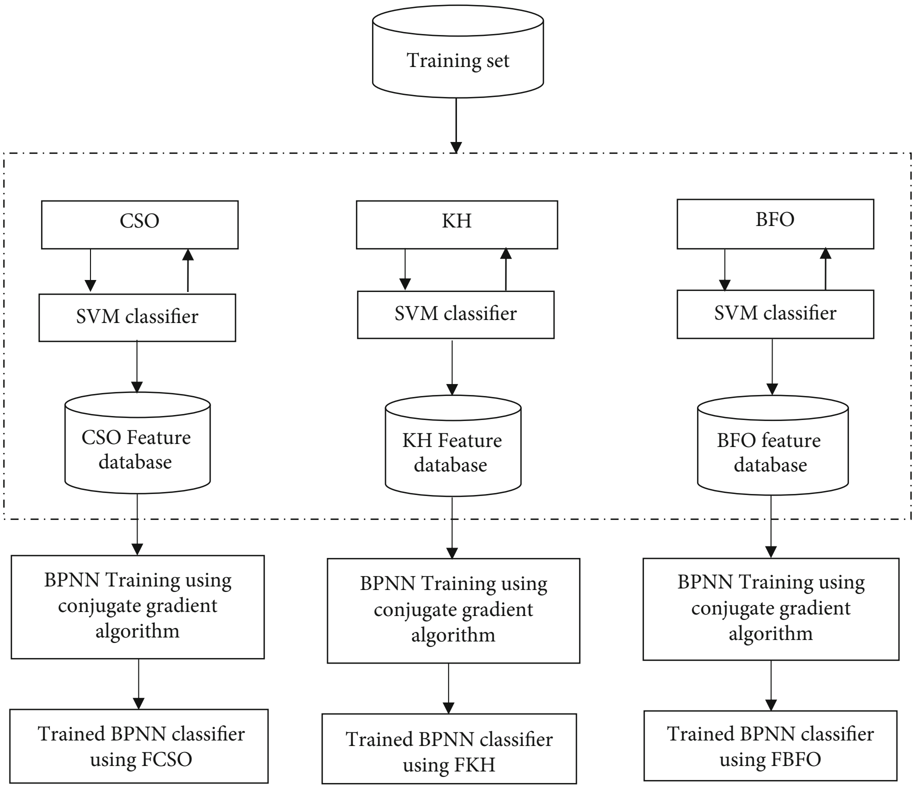
Training set
CSO
KH
BFO
SVM classifier
SVM classifier
SVM classifier
CSO Feature database
KH Feature database
BFO feature database
BPNN Training using conjugate gradient algorithm
BPNN Training using conjugate gradient algorithm
BPNN Training using conjugate gradient algorithm
Trained BPNN classifier using FCSO
Trained BPNN classifier using FKH
Trained BPNN classifier using FBFO

Figure 2: Classification using BPNN.

Table 7: Parameter settings for BPNN.

BPNN parameter	Bioinspired algorithm	WDBC dataset	SHD dataset	HCC dataset	HD dataset	VCD dataset	CHD dataset	ILP dataset
Number of input nodes	CSO	15	9	20	16	3	6	5
	KH	17	10	39	10	3	10	8
	BFO	18	9	35	19	2	11	5
Number of hidden nodes	CSO	30	18	40	32	6	12	10
	KH	34	20	78	20	6	20	16
	BFO	36	18	70	38	4	22	10

clinical dataset C_s is subject to min-max normalization, the minimum value ($\min_A$) and maximum value ($\max_A$) in the value set of "A" are first identified, and normalization is performed using the formula presented in equation (1).

$$a' = \frac{a - \min_A}{\max_A - \min_A}\left(new_{\max_A} - new_{\min_A}\right) + new_{\min_A}. \quad (1)$$

If the formula "a'" is the normalized value of an attribute "a," when a is drawn from the value set of "A." Since min-max normalization is being used to normalize the values in the range 0 to 1, the value of $\text{new}_{\max_A}$ is 1 and $\text{new}_{\min_A}$ is 0.

The number of instances in each C_s used for constructing and testing the classifier prior to generating additional samples using SMOTE, the number of instances in each C_s after generating additional samples using SMOTE, the number of instances in the training set (T_s), and the number of instances in the testing set (T_t) is presented in Table 3. After preprocessing, each C_s is split into training set (60%) and testing set (40%).

5.2. Feature Selection. Feature selection is performed on each T_s used for experimentation to select the optimal features for training the classifier. Selecting the optimal features from the T_s will improve the classification accuracy. A wrapper approach that uses three bioinspired algorithms, namely, CSO, KH, and BFO with the accuracy of the SVM classifier is used to perform feature selection. An outline of CSO, KH, and BFO used for feature selection is presented below.

5.2.1. Outline of the CSO Algorithm for Feature Selection. CSO is inspired and modeled based on two main postures of cats, namely, resting and tracing. Mimicking the resting behavior of a cat is named as seeking mode, and mimicking the tracing behavior of a cat is named as tracing mode. The seeking mode relates to a local search process, whereas the

```
Input: training set (FCSO, FKH, FBFO).
Step 1: initialize the parameters, namely, weights and bias, number of hidden layers, and learning rate of the BPNN.
Step 2: the number of hidden nodes are calculated using Equation (18).
$H = 2n$.(18)
In the above formula, $H$ is the number of hidden nodes, and $n$ is the number of input nodes.
Step 3: the input of the hidden layer is calculated using Equation (19).
$I_h = \sum w_{ij} O_j + \varnothing_j$.(19)
In the above formula, $I_h$ is the input of the hidden layer; $w_{ij}$ is the weights of each input nodes; $\varnothing_j$ is the bias.
Step 4: the output of the hidden layer is calculated using Equation (20).
$O_j = 1/1 + e^{-I_j}$,(20)
where $O_j$ is the output of the $j^{th}$ hidden layer, and $I_j$ is the input to the neuron from the previous layer.
Step 5: calculate the error rate in the predicted output using Equation (21).
$\text{Error Rate (MSE)} = \sum_{i=1}^{0} (Z_i^k - C_i^k)^2$.(21)
In the above formula, $Z_i^k$ is the expected output, and $C_i^k$ is the obtained output.
Step 6: update the new weights and bias based on the learning rate and error rate using CGA.
Step 7: repeat the steps from 2 to 5 until the error rate converges.
Output: three BPNN classifiers trained using FCSO, FKH, and FBFO.
```

Algorithm 4:

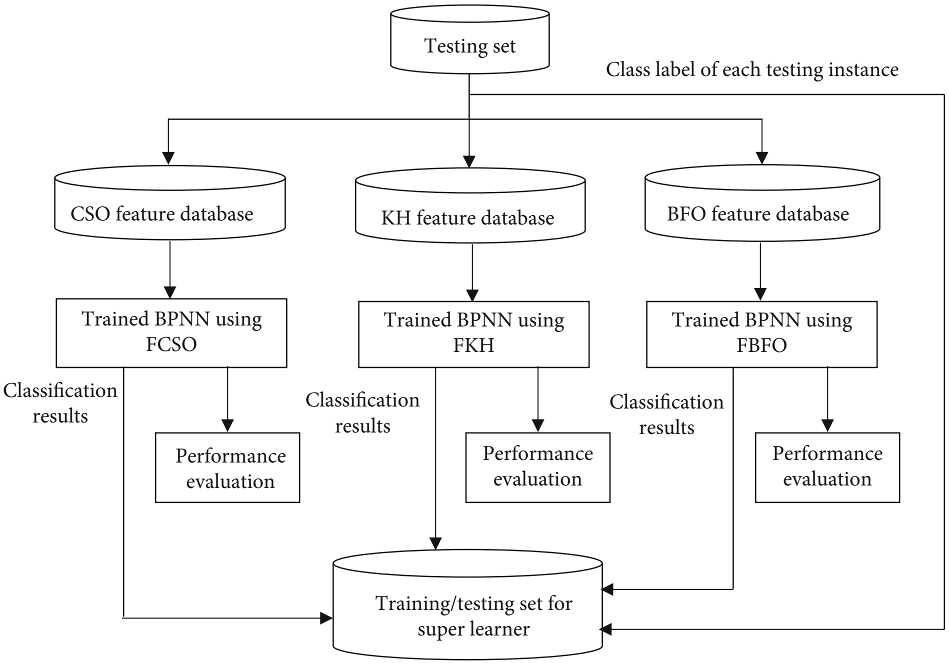

Figure 3: Classifier testing.

tracing mode relates to a global search process. The vital parameters that play an important role in CSO are outlined in Table 4. Tracing mode relates to cat's movement while chasing a prey, for example, chasing a rat.

The steps to select the optimal feature subset using CSO is outlined below (Algorithm 1):

5.2.2. Outline of the KH Algorithm for Feature Selection. The KH algorithm is centered on the imitation of the herding behavior of krill swarms with respect to precise biological and environmental processes. Krill density is reduced by predators, namely, seals, penguins, or seabirds. The herding of the krill individuals includes, one, increasing the krill density and two, reaching the food. The fitness function of each krill individual has been defined as the least distance of each individual krill from food and from the highest density of the herd.

Three vital actions considered to define the time-dependent position of an individual krill are one, movement induced by other krill individuals, two, foraging activity, and three, random diffusion.

Krill individuals attempt to maintain a high density and hence move due to their mutual effect. Local swarm density, target swarm density, and repulsive swarm density are used

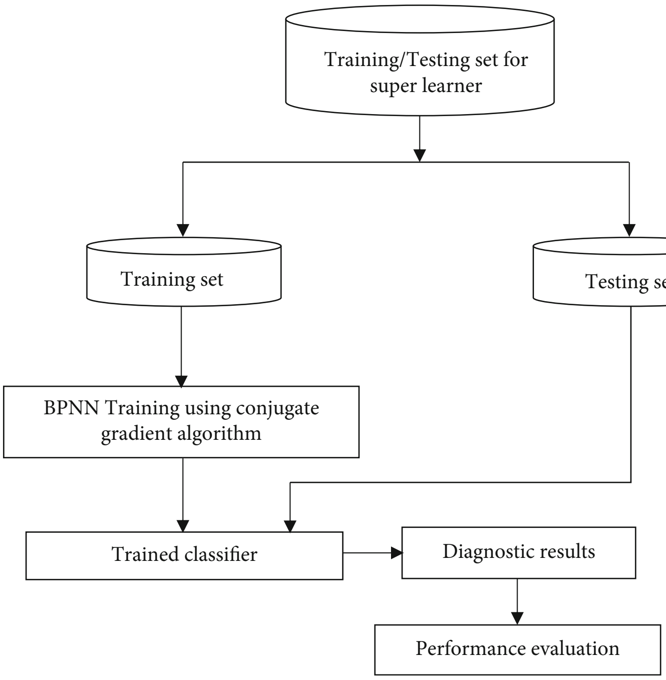
Training/Testing set for super learner
Training set
Testing set
BPNN Training using conjugate gradient algorithm
Trained classifier
Diagnostic results
Performance evaluation

Figure 4: Super learner training and testing.

Table 8: Parameter settings for super learner.

Name of the parameter	WDBC dataset	SHD dataset	HCC dataset	HD dataset	VCD dataset	CHD dataset	ILP dataset
Initial population size	250	86	73	80	131	97	240
Number of input nodes	3	3	3	3	3	3	3
Number of hidden nodes	6	6	6	6	6	6	6

Table 9: Performance of FCSO, FKH, and FBFO classifiers and super learner on WDBC dataset.

Feature selection algorithm	Size of feature subset	TN	FP	FN	TP	Accuracy	Sensitivity	Specificity	Precision	*F*-score
CSO	15	137	4	6	165	96.79	96.49	97.16	97.63	0.97
KH	17	139	2	5	166	97.76	97.08	98.58	98.81	0.98
BFO	18	139	2	8	163	96.79	95.32	98.58	98.79	0.97
Super learner	—	22	0	2	39	96.83	95.12	100.00	100.00	0.98

Table 10: Performance of FCSO, FKH, and FBFO classifiers and super learner on Statlog dataset.

Feature selection algorithm	Size of feature subset	TN	FP	FN	TP	Accuracy	Sensitivity	Specificity	Precision	*F*-score
CSO	9	53	8	9	38	84.26	80.85	86.89	82.61	0.82
KH	10	53	8	11	36	82.41	76.60	86.89	81.82	0.79
BFO	9	51	10	10	37	81.48	78.72	83.61	78.72	0.79
Super learner	—	10	1	2	9	86.36	81.82	90.91	90.00	0.86

Table 11: Performance of FCSO, FKH, and FBFO classifiers and super learner on HCC dataset.

Feature selection algorithm	Size of feature subset	TN	FP	FN	TP	Accuracy	Sensitivity	Specificity	Precision	*F*-score
CSO	20	43	9	9	31	80.43	77.50	82.69	77.50	0.78
KH	39	48	4	13	27	81.52	67.50	92.31	87.10	0.76
BFO	35	47	5	20	20	72.83	50.00	90.38	80.00	0.62
Super learner	—	10	1	0	8	94.74	100.00	90.91	88.89	0.94

Table 12: Performance of FCSO, FKH, and FBFO classifiers and super learner on hepatitis dataset.

Feature selection algorithm	Size of feature subset	TN	FP	FN	TP	Accuracy	Sensitivity	Specificity	Precision	F-score
CSO	16	47	2	10	42	88.12	80.77	95.92	95.45	0.88
KH	10	45	4	6	46	90.10	88.46	91.84	92.00	0.90
BFO	19	47	2	12	40	86.14	76.92	95.92	95.24	0.85
Super learner	—	8	1	1	11	90.48	91.67	88.89	91.67	0.92

Table 13: Performance of FCSO, FKH, and FBFO classifiers and super learner on vertebral column dataset.

Feature selection algorithm	Size of feature subset	TN	FP	FN	TP	Accuracy	Sensitivity	Specificity	Precision	F-score
CSO	3	74	10	17	63	83.54	78.75	88.10	86.30	0.82
KH	3	81	3	13	67	90.24	83.75	96.43	95.71	0.89
BFO	2	80	4	17	63	87.20	78.75	95.24	94.03	0.86
Super learner	—	19	2	4	8	81.82	66.67	90.48	80.00	0.73

Table 14: Performance of FCSO, FKH, and FBFO classifiers and super learner on Cleveland heart disease dataset.

Feature selection algorithm	Size of feature subset	TN	FP	FN	TP	Accuracy	Sensitivity	Specificity	Precision	F-score
CSO	6	60	8	12	42	83.61	77.78	88.24	84.00	0.81
KH	10	56	12	11	43	81.15	79.63	82.35	78.18	0.79
BFO	11	53	15	13	41	77.05	75.93	77.94	73.21	0.75
Super learner	—	13	2	2	8	84.00	80.00	86.67	80.00	0.80

Table 15: Performance of FCSO, FKH, and FBFO classifiers and super learner on Indian liver patient dataset.

Feature selection algorithm	Size of feature subset	TN	FP	FN	TP	Accuracy	Sensitivity	Specificity	Precision	F-score
CSO	5	103	62	33	102	68.33	75.56	62.42	62.20	0.68
KH	8	104	61	40	95	66.33	70.37	63.03	60.90	0.65
BFO	5	101	64	34	101	67.33	74.81	61.21	61.21	0.67
Super learner	—	26	15	3	16	70.00	84.21	63.41	51.61	0.64

to estimate the direction of motion. Food location and prior experience about the food location are the two parameters used to estimate the foraging motion. Random diffusion is used for the exploration of the search space. In the KH algorithm, the population diversity is improved by means of the diffusion function, which is integrated into the krill individuals. Random diffusion is the net movement of each krill individual from high-density to low-density regions.

The motion velocity of krill particle applies the Lagrangian model [43] as shown in Equation (5).

$$\frac{dx_i}{dt} = N_i + F_i + \mathrm{RD}_i. \quad (5)$$

In the above formula, dx_i/dt is the motion velocity of krill particle i, N_i is the induced motion, F_i is the foraging motion, and RD_i is the random diffusion of the i^{th} krill individual. The vital parameters that play an important role in the KH algorithm are outlined in Table 5.

The steps to select the optimal feature subset using KH is outlined below (Algorithm 2):

5.2.3. Outline of the BFO Algorithm for Feature Selection. The bacterial foraging optimization (BFO) algorithm imitates the pattern exhibited during the foraging process of Escherichia coli bacteria, that includes chemotaxis, swarming, reproduction, and elimination-dispersal operations [41]. The basic idea behind the foraging strategy of E. coli bacteria is to obtain the maximum nutrition in a unit time. The chemotaxis strategy involves the searching of nutrition by taking small movements such as tumbling, moving, and swimming, using its locomotory organ called flagella. The swarming strategy deals with the communication between bacteria. When the bacteria discover high amount of nutrients, they will release chemical substances to attract other bacteria. If they are in danger, they will tend to prevent other bacteria. The reproduction process involves splitting of healthier bacterium into two bacteria, and the low healthy bacteria are set to die. Finally, the elimination-dispersal strategy involves replacing the low health bacterium by randomly generated new ones. The vital parameters that play an important role in the BFO algorithm are outlined in Table 6.

The steps involved in finding the optimal feature subset using the BFO algorithm is outlined below:

Table 16: Comparison of the proposed work and existing work using clinical dataset.

Author/year	Method/reference	Accuracy %						
		WDBC	SHD	HCC	HD	VCD	CHD	ILP
Ayon et al. (2020)	DNN [45]	—	98.15	—	—	—	94.39	—
	SVM [45]	—	97.41	—	—	—	97.36	—
Bai Ji et al. (2020)	IBPSO with k-NN [46]	96.14	—	—	—	—	—	—
Elgin et al. (2020)	Cooperative coevolution and RF [30]	97.1	96.8	72.2	82.3	91.4	93.4	—
Magesh et al. (2020)	Cluster-based decision tree [47]	—	—	—	—	—	89.30	—
Rabbi et al. (2020)	PCC and AdaBoost [48]	—	—	—	—	—	—	92.19
Rajesh et al. (2020)	RF classifier [49]	—	—	80.64	—	—	—	—
Salima et al. (2020)	ECSA with k-NN [50]	95.76	82.96	—	—	—	—	—
Singh J et al. (2020)	Logistic regression [51]	—	—	—	—	—	—	74.36
Sreejith et al. (2020)	CMVO and RF [28]	—	—	—	—	—	—	82.46
Sreejith et al. (2020)	DISON and ERT[27]	—	94.5	—	—	87.17	93.67	—
Tougui et al. (2020)	ANN with Matlab [52]	—	—	—	—	—	85.86	—
Tubishat et al. (2020)	ISSA with k-NN [53]	—	88.1	—	—	89.0	—	—
Abdar et al. (2019)	Novel nested ensemble nu-SVC [54]	—	—	—	—	—	98.60	—
Anter et al. (2019)	CFCSA with chaotic maps [31]	98.6	—	—	68.0	—	88.0	68.4
Aouabed et al. (2019)	Nested ensemble nu-SVC, GA and multilevel balancing [55]	—	—	—	—	—	98.34	—
Elgin et al. (2019)	DE, LO and GSO with Adaboost SVM [32]	98.73	—	—	93.9	—	—	—
Książek et al. (2019)	SVM [56]	—	97.41	—	—	—	97.36	—
Sayed et al. (2019)	Novel chaotic crow search algorithm with k-NN [57]	90.28	78.84	—	83.7	—	—	71.68
Abdar et al. (2018)	MPNN and C5.0 [58]	—	—	—	—	—	—	94.12
Abdullah et al. (2018)	k-NN [59]	—	—	—	—	85.32	—	—
	RF [59]	—	—	—	—	79.57	—	—
Sawhney et al. (2018)	BFA and RF [60]	—	—	83.50	—	—	—	—
Abdar et al. (2017)	Boosted C5.0 [61]	—	—	—	—	—	—	93.75
	CHAID [61]	—	—	—	—	—	—	65.0
Zamani et al. (2016)	WOA with k-NN [62]	—	77.05	—	87.10	—	—	—
Abdar (2015)	SVM with rapid miner [63]	—	—	—	—	—	—	72.54
	C5.0 with IBM SPSS modeller [63]	—	—	—	—	—	—	87.91
Santos et al. (2015)	Neural networks and augmented set approach [64]	—	—	75.2	—	—	—	—
Chiu et al. (2013)	ANN and LR [65]	—	—	85.10	—	—	—	—
Mauricio et al. (2013)	ABCO with SVM [66]	—	84.81	—	87.10	—	83.17	—
Proposed	CSO, KH, BFO, and super learner	96.83	86.36	94.74	90.48	81.82	84.00	70.00

5.3. Classifier Training. Each C_s is preprocessed and split into training set (T_s – 60%) and testing set (T_t – 40%). A wrapper approach that uses three bioinspired algorithms CSO, KH, and BFO with the classification accuracy of SVM as the fitness function has been used for feature selection. The features selected by each bioinspired algorithm are used to train three BPNNs independently using CGA. The number of hidden layers for each BPNN is 1, and the activation function used in the hidden layer is sigmoid. The learning rate is $1e$–07, and the maximum number of iterations is 100. Since the classification is binary, each BPNN has only one output node, and the activation function used in the output layer is sigmoid. Figure 2 elaborates the process of training BPNN classifiers.

The number of training instances for FCSO, FKH, and FBFO classifiers is presented in Table 3. Though majority of the features selected by each bioinspired algorithm overlap, it has been inferred that the number of features selected by each algorithm is not the same. The parameter settings for each classifier is presented in Table 7.

The steps to train the BPNN classifier using three BPNN classifier and trained using CSO, KH, and BFO algorithms are outlined below:

5.4. Classifier Testing and Dataset Construction for Super Learner. After training the classifier with 60% of the preprocessed C_s (T_s), classifier testing is performed using the remaining 40% of the of the preprocessed C_s (T_t). Figure 3 elaborates the process of testing the three classifiers and also throws light on the process of training the super learner.

Feature selection is performed on the testing set by querying the FCSO, FKH, and FBFO databases. The instances of the testing set containing the features selected by the CSO are used to test the FCSO classifier; similarly, the instances of the testing set containing the features selected by the KH and BFO are used to test the FKH and FBFO classifier. The performance of the FCSO, FKH, and FBFO classifiers are evaluated using the results obtained from the testing set.

The classification result of each instance of the testing set for FCSO, FKH, and FBFO classifiers and the class label corresponding to each instance of the testing set will be the candidate instances for training and testing the super learner.

5.5. Super Learner Training and Testing. As outlined in Section 5.4, the classification result pertaining to each instance of the testing set for FCSO, FKH, and FBFO classifiers and the class label corresponding to each instance of the testing set will be the candidate instances for training and testing the super learner. Figure 4 elaborates the process of training and testing of the super learner. The training set comprises of 80% of the instances, and the testing set comprises of 20% of the instances. The number of training and testing instances for the super learner is presented in Table 3.

Super learner is a type of ensemble classifier [44]. In this work, a BPNN classifier trained using CGA is used as the super learner. The parameter settings for the super learner are presented in Table 8.

The super learner is trained using the steps presented in Section 5.3 for training the BPNN classifier using CGA, and the performance of the super learner is evaluated using the testing set.

6. Results and Discussions

Seven clinical datasets from the UCI ML repository, namely, WDBC, SHD, HCC, HD, VCD, CHD, and ILP have been used for experimentation. The performance of the FCSO, FKH, and FBFO classifiers and super learner is evaluated in terms of accuracy, sensitivity, specificity, precision, and F -score, which are calculated based on true positive (TP), true negative (TN), false positive (FP), and false negative (FN) using Equations (22), (23), (24), (25), and (26).

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \tag{22}$$

In the above formula, TP is the number of positive instances predicted as positive by the classifier, TN is the number of negative instances predicted as negative by the classifier, FP is the number of negative instances predicted as positive by the classifier, and FN is the number of positive instances predicted as negative by the classifier.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{23}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}, \tag{24}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{25}$$

$$\text{F} - \text{score} = \frac{2\text{TP}}{2\text{TP} + (\text{FP} + \text{FN})}. \tag{26}$$

Accuracy, sensitivity, specificity, precision, and $F-$ score obtained using FCSO, FKH, and FBFO classifiers and super learner for the datasets WDBC, SHD, HCC, HD, VCD, CHD, and ILP are presented in Tables 9–15.

The super learner has achieved a classification accuracy of 96.83% for WDBC, 86.36% for SHD, 94.74% for HCC, 90.48% for HD, 81.82% for VCD, 84.0% for CHD, and 70.0% for ILP. The classification accuracy of the proposed work has been compared with the performance of the existing work on clinical datasets and the comparison results summarized in Table 16.

7. Conclusion and Scope for Future Work

A CAD system that employs a super learner to diagnose the presence or absence of a disease has been implemented in this work. Seven C_s from the UCI ML repository, namely, WDBC, SHD, HCC, HD, VCD, CHD, and ILP have been used for experimentation. Each C_s is preprocessed, and the preprocessed C_s is split into training and testing sets. A wrapper-based feature selection approach using three bioinspired algorithms, namely, CSO, KH, and BFO, with the accuracy of SVM classifier has been used to select the optimal feature subsets. The selected feature subsets are used to train three BPNN classifiers using CGA, and the performance of the trained classifiers is evaluated. The classification results obtained for each instance of the testing set of the three classifiers and the class label associated with each instance of the testing set will be the candidate instances for training and testing the super learner. The super learner achieved a classification accuracy of 96.83% for WDBC, 86.36% for SHD, 94.74% for HCC, 90.48% for HD, 81.82% for VCD, 84.0% for CHD, and 70.0% for ILP.

CAD systems to diagnose disorders in the human body from different imaging modalities such as X-ray, computed tomography, magnetic resonance imaging, and positron emission tomography are gaining importance. This work can be extended by developing CAD systems to diagnose disorders from the medical images acquired through different imaging modalities. Features based on shape, texture, and run length can be extracted from the images, and the feature selection algorithms used in this work can be used to select the relevant features. The relevant features can be used to build classifier models to predict the presence or absence of disorders from the images.

Data Availability

The data supporting this study are from previously reported studies and datasets, which have been cited. The datasets used in this research work are available at UCI Machine Learning Repository.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[2] R. C. Prati, "Combining feature ranking algorithms through rank aggregation," in *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, Brisbane, QLD, Australia, 2012.

[3] A. G. Karegowda, A. S. Manjunath, and M. A. Jayaram, "Comparative study of attribute selection using gain ratio and correlation based feature selection," *International Journal of Information Technology and Knowledge Management*, vol. 2, pp. 271–277, 2010.

[4] W. Shang, H. Huang, H. Zhu, Y. Lin, Y. Qu, and Z. Wang, "A novel feature selection algorithm for text categorization," *Expert Systems with Applications*, vol. 33, no. 1, pp. 1–5, 2007.

[5] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," *Advances in Neural Information Processing Systems*, vol. 18, pp. 507–514, 2005.

[6] A. Suebsing and H. Nualsawat, "A novel technique for feature subset selection based on cosine similarity," *Applied Mathematical Sciences*, vol. 6, pp. 6627–6655, 2012.

[7] K. B. Nahato, K. N. Harichandran, and K. Arputharaj, "Knowledge mining from clinical datasets using rough sets and backpropagation neural network," *Computational and Mathematical Methods in Medicine*, vol. 2015, Article ID 460189, 13 pages, 2015.

[8] J. J. Christopher, H. K. Nehemiah, K. Arputharaj, and G. L. Moses, "Computer-assisted medical decision-making system for diagnosis of Urticaria," *MDM Policy & Practice*, vol. 1, no. 1, 2016.

[9] D. S. Elizabeth, C. S. Retmin Raj, H. K. Nehemiah, and A. Kannan, "Computer-aided diagnosis of lung cancer based on analysis of the significant slice of chest computed tomography image," *IET Image Processing*, vol. 6, no. 6, pp. 697–705, 2012.

[10] D. S. Elizabeth, H. K. Nehemiah, C. S. R. Raj, and A. Kannan, "A novel segmentation approach for improving diagnostic accuracy of CAD systems for detecting lung cancer from chest computed tomography images," *Journal of Data and Information Quality (JDIQ)*, vol. 3, pp. 1–16, 2012.

[11] D. S. Elizabeth, A. Kannan, and H. K. Nehemiah, "Computer aided diagnosis system for the detection of bronchiectasis in chest computed tomography images," *International Journal of Imaging Systems and Technology*, vol. 19, no. 4, pp. 290–298, 2009.

[12] S. E. Darmanayagam, K. N. Harichandran, S. R. R. Cyril, and K. Arputharaj, "A novel supervised approach for segmentation of lung parenchyma from chest CT for computer-aided diagnosis," *Journal of Digital Imaging*, vol. 26, no. 3, pp. 496–509, 2013.

[13] C. S. Retmin Raj, H. K. Nehemiah, D. S. Elizabeth, and A. Kannan, "A novel feature-significance based k-nearest neighbour classification approach for computer aided diagnosis of lung disorders," *Current Medical Imaging Reviews*, vol. 14, no. 2, pp. 289–300, 2018.

[14] A. Titus, H. K. Nehemiah, and A. Kannan, "Classification of interstitial lung diseases using particle swarm optimized support vector machine," *International Journal of Soft Computing*, vol. 10, pp. 25–36, 2015.

[15] Y. Nancy Jane, H. Khanna Nehemiah, and K. Arputharaj, "A Q-backpropagated time delay neural network for diagnosing severity of gait disturbances in Parkinson's disease," *Journal of Biomedical Informatics*, vol. 60, pp. 169–176, 2016.

[16] J. Y. Nancy, N. H. Khanna, and A. Kannan, "A bio-statistical mining approach for classifying multivariate clinical time series data observed at irregular intervals," *Expert Systems with Applications*, vol. 78, pp. 283–300, 2017.

[17] N. Leema, H. Khanna Nehemiah, A. Kannan, and J. Jabez Christopher, "Computer aided diagnosis system for clinical decision making: experimentation using Pima Indian diabetes dataset," *Asian Journal of Information Technology*, vol. 15, pp. 3217–3231, 2016.

[18] K. B. Nahato, K. H. Nehemiah, and A. Kannan, "Hybrid approach using fuzzy sets and extreme learning machine for classifying clinical datasets," *Informatics in Medicine Unlocked*, vol. 2, pp. 1–11, 2016.

[19] J. J. Christopher, H. K. Nehemiah, and A. Kannan, "A swarm optimization approach for clinical knowledge mining," *Computer Methods and Programs in Biomedicine*, vol. 121, no. 3, pp. 137–148, 2015.

[20] N. Leema, H. K. Nehemiah, and A. Kannan, "Neural network classifier optimization using differential evolution with global information and back propagation algorithm for clinical datasets," *Applied Soft Computing*, vol. 49, pp. 834–844, 2016.

[21] N. Leema, H. K. Nehemiah, and A. Kannan, "Quantum-behaved particle swarm optimization based radial basis function network for classification of clinical datasets," *International Journal of Operations Research and Information Systems (IJORIS)*, vol. 9, pp. 32–52, 2020.

[22] J. Y. Nancy, N. H. Khanna, and K. Arputharaj, "Imputing missing values in unevenly spaced clinical time series data to build an effective temporal classification framework," *Computational Statistics & Data Analysis*, vol. 112, pp. 63–79, 2017.

[23] N. Y. Jane, K. Nehemiah, and K. Arputharaj, "A temporal mining framework for classifying un-evenly spaced clinical data: an approach for building effective clinical decision-making system," *Applied Clinical Informatics*, vol. 7, no. 1, pp. 1–21, 2016.

[24] I. Fister Jr., X. S. Yang, I. Fister, J. Brest, and Fister, "A Brief Review of Nature-Inspired Algorithms for Optimization," *Elektrotehniski Vestnik*, vol. 80, no. 3, pp. 1–7, 2013.

[25] N. Leema, K. H. Nehemiah, V. R. Elgin Christo, and A. Kannan, "Evaluation of parameter settings for training neural networks using bBackpropagation algorithms," *International Journal of Operations Research and Information Systems*, vol. 11, no. 4, pp. 62–85, 2020.

[26] V. R. E. Christo, H. K. Nehemiah, K. B. Nahato, J. Brighty, and A. Kannan, "Computer assisted medical decision-making system using genetic algorithm and extreme learning machine for diagnosing allergic rhinitis," *International Journal of Bio-Inspired Computation*, vol. 16, no. 3, pp. 148–157, 2020.

[27] S. Sreejith, H. Khanna Nehemiah, and A. Kannan, "A classification framework using a diverse intensified strawberry optimized neural network (DISON) for clinical decision-making," *Cognitive Systems Research*, vol. 64, pp. 98–116, 2020.

[28] S. Sreejith, H. Khanna Nehemiah, and A. Kannan, "Clinical data classification using an enhanced SMOTE and chaotic evolutionary feature selection," *Computers in Biology and Medicine*, vol. 126, p. 103991, 2020.

[29] A. Isaac, H. K. Nehemiah, A. Isaac, and A. Kannan, "Computer-aided diagnosis system for dDiagnosis of pulmonary emphysema using bio-inspired aAlgorithms," *Computers in Biology and Medicine*, vol. 124, p. 103940, 2020.

[30] V. E. Christo, H. K. Nehemiah, J. Brighty, and A. Kannan, "Feature selection and instance selection from clinical datasets using co-operative co-evolution and classification using random Forest," *IETE Journal of Research*, pp. 1–14, 2020.

[31] A. M. Anter and 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 Computing*, vol. 24, no. 3, pp. 1565–1584, 2020.

[32] V. R. Elgin Christo, H. Khanna Nehemiah, B. Minu, and A. Kannan, "Correlation-based ensemble feature selection using bioinspired algorithms and classification Using backpropagation neural network," *Computational and mathematical methods in medicine*, vol. 2019, Article ID 7398307, 17 pages, 2019.

[33] J. D. Sweetlin, H. K. Nehemiah, and A. Kannan, "Computer aided diagnosis of drug sensitive pulmonary tuberculosis with cavities, consolidations and nodular manifestations on lung CT images," *International Journal of Bio-Inspired Computation*, vol. 13, no. 2, pp. 71–85, 2019.

[34] J. Dhalia Sweetlin, H. K. Nehemiah, and A. Kannan, "Computer aided diagnosis of pulmonary hamartoma from CT scan images using ant colony optimization based feature selection," *Alexandria Engineering Journal*, vol. 57, no. 3, pp. 1557–1567, 2018.

[35] J. D. Sweetlin, H. K. Nehemiah, and A. Kannan, "Feature selection using ant colony optimization with tandem-run recruitment to diagnose bronchitis from CT scan images," *Computer Methods and Programs in Biomedicine*, vol. 145, pp. 115–125, 2017.

[36] C. Sunil Retmin Raj, H. Khanna Nehemiah, D. Shiloah Elizabeth, and A. Kannan, "Distance based genetic algorithm for feature selection in computer aided diagnosis systems," *Current Medical Imaging Reviews*, vol. 13, no. 3, pp. 284–298, 2017.

[37] H. M. Zawbaa, E. Emary, B. Parv, and M. Sharawi, "Feature selection approach based on moth-flame optimization algorithm," in *2016 IEEE Congress on Evolutionary Computation (CEC)*, pp. 4612–4617, Vancouver, BC, Canada, July 2016.

[38] S.-C. Chu, P.-W. Tsai, and J.-S. Pan, "Cat swarm optimization," in *PRICAI 2006: Trends in Artificial Intelligence*, pp. 854–858, Springer, 2006.

[39] A. H. Gandomi and A. H. Alavi, "Krill herd: a new bio-inspired optimization algorithm," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 12, pp. 4831–4845, 2012.

[40] H. Chen, Y. Zhu, and K. Hu, "Cooperative bacterial foraging optimization," *Discrete Dynamics in Nature and Society*, vol. 2009, Article ID 815247, 17 pages, 2009.

[41] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Systems Magazine*, vol. 22, no. 3, pp. 52–67, 2002.

[42] H. Chen, Y. Zhu, and K. Hu, "Adaptive bacterial foraging optimization," *Abstract and Applied Analysis*, vol. 2011, Article ID 108269, 27 pages, 2011.

[43] E. E. Hofmann, A. G. E. Haskell, J. M. Klinck, and C. M. Lascara, "Lagrangian modelling studies of antarctic krill (Euphausia superba) swarm formation," *ICES Journal of Marine Science*, vol. 61, no. 4, pp. 617–631, 2004.

[44] M. J. van der Laan, E. C. Polley, and A. E. Hubbard, "Super learner," *Statistical Applications in Genetics and Molecular Biology*, vol. 6, no. 1, p. 25, 2007.

[45] S. I. Ayon, M. M. Islam, and M. R. Hossain, "Coronary artery heart disease prediction: a comparative study of computational intelligence techniques," *IETE Journal of Research*, pp. 1–20, 2020.

[46] B. Ji, X. Lu, G. Sun, W. Zhang, J. Li, and Y. Xiao, "Bio-inspired feature selection: an improved binary particle swarm optimization approach," *IEEE Access*, vol. 8, pp. 85989–86002, 2020.

[47] G. Magesh and P. Swarnalatha, "Optimal feature selection through a cluster-based DT learning (CDTL) in heart disease prediction," *Evolutionary Intelligence*, pp. 1–11, 2020.

[48] M. F. Rabbi, S. M. Hasan, A. I. Champa, M. Asif Zaman, and M. K. Hasan, "Prediction of liver disorders using machine learning algorithms: a comparative study," in *2020 2nd International Conference on Advanced Information and Communication Technology (ICAICT)*, pp. 111–116, Dhaka, Bangladesh, November 2020.

[49] S. Rajesh, N. A. Choudhury, and S. Moulik, "Hepatocellular carcinoma (HCC) liver cancer prediction using machine learning algorithms," in *2020 IEEE 17th India Council International Conference (INDICON)*, pp. 1–5, New Delhi, India, December 2020.

[50] S. Ouadfel and M. Abd Elaziz, "Enhanced crow search algorithm for feature selection," *Expert Systems with Applications*, vol. 159, p. 113572, 2020.

[51] J. Singh, S. Bagga, and R. Kaur, "Software-based prediction of liver disease with feature selection and classification techniques," *Procedia Computer Science*, vol. 167, pp. 1970–1980, 2020.

[52] I. Tougui, A. Jilbab, and J. el Mhamdi, "Heart disease classification using data mining tools and machine learning techniques," *Health and Technology*, vol. 10, no. 5, pp. 1137–1144, 2020.

[53] M. Tubishat, N. Idris, L. Shuib, M. A. M. Abushariah, and S. Mirjalili, "Improved Salp swarm algorithm based on opposition based learning and novel local search algorithm for feature selection," *Expert Systems with Applications*, vol. 145, p. 113122, 2020.

[54] M. Abdar, U. R. Acharya, N. Sarrafzadegan, and V. Makarenkov, "NE-nu-SVC: a new nested ensemble clinical decision support system for effective diagnosis of coronary artery disease," *IEEE Access*, vol. 7, pp. 167605–167620, 2019.

[55] Z. Aouabed, M. Abdar, N. Tahiri, J. C. Gareau, and V. Makarenkov, "A novel effective ensemble model for early detection of coronary artery disease," in *International Conference Europe Middle East & North Africa Information Systems and Technologies to Support Learning*, pp. 480–489, Springer, 2019.

[56] W. Książek, M. Abdar, U. R. Acharya, and P. Pławiak, "A novel machine learning approach for early detection of

hepatocellular carcinoma patients," *Cognitive Systems Research*, vol. 54, pp. 116–127, 2019.

[57] G. I. Sayed, A. E. Hassanien, and A. T. Azar, "Feature selection via a novel chaotic crow search algorithm," *Neural Computing and Applications*, vol. 31, no. 1, pp. 171–188, 2019.

[58] M. Abdar, N. Y. Yen, and J. C. S. Hung, "Improving the diagnosis of liver disease using multilayer perceptron neural network and boosted decision trees," *Journal of Medical and Biological Engineering*, vol. 38, no. 6, pp. 953–965, 2018.

[59] A. A. Abdullah, A. Yaakob, and Z. Ibrahim, "Prediction of spinal abnormalities using machine learning techniques," in *2018 International Conference on Computational Approach in Smart Systems Design and Applications (ICASSDA)*, pp. 1–6, Kuching, Malaysia, August 2018.

[60] R. Sawhney, P. Mathur, and R. Shankar, "A firefly algorithm based wrapper-penalty feature selection method for cancer diagnosis," in *Computational Science and Its Applications – ICCSA 2018*, vol. 10960, pp. 438–449, Springer, 2018.

[61] M. Abdar, M. Zomorodi-Moghadam, R. Das, and I. H. Ting, "Performance analysis of classification algorithms on early detection of liver disease," *Expert Systems with Applications*, vol. 67, pp. 239–251, 2017.

[62] H. Zamani and M. H. Nadimi-Shahraki, "Feature selection based on whale optimization algorithm for diseases diagnosis," *International Journal of Computer Science and Information Security*, vol. 14, pp. 1243–1247, 2016.

[63] M. ABDAR, "A survey and compare the performance of IBM SPSS modeler and rapid miner software for predicting liver disease by using various data mining algorithms," *Cumhuriyet Üniversitesi Fen-Edebiyat Fakültesi Fen Bilimleri Dergisi*, vol. 36, pp. 3230–3241, 2015.

[64] M. S. Santos, P. H. Abreu, P. J. García-Laencina, A. Simão, and A. Carvalho, "A new cluster-based oversampling method for improving survival prediction of hepatocellular carcinoma patients," *Journal of Biomedical Informatics*, vol. 58, pp. 49–59, 2015.

[65] H.-C. Chiu, T.-W. Ho, K.-T. Lee, H.-Y. Chen, and W.-H. Ho, "Mortality predicted accuracy for hepatocellular carcinoma patients with hepatic resection using artificial neural network," *The Scientific World Journal*, vol. 2013, Article ID 201976, 10 pages, 2013.

[66] M. Schiezaro and H. Pedrini, "Data feature selection based on artificial bee colony algorithm," *EURASIP Journal on Image and Video Processing*, vol. 2013, no. 1, Article ID 47, 2013.