A New Hybrid Intelligent System for Prediction of Medical Diseases

Sultan Noman Qasem[1,2]
[1]Computer Science Department, College of Computer and Information Sciences
Al Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Saudi Arabia
[2]Computer Science Department, Faculty of Applied Science
Taiz University, Taiz, Yemen

Monirah Alsaidan
Computer Science Department, College of Computer and Information Sciences
Al Imam Mohammad Ibn Saud Islamic University (IMSIU)
Riyadh, Saudi Arabia

***Abstract*—This paper proposes a hybrid intelligent system as medical decision support tool for data classification based on the Neural Network, Galactic Swarm Optimization (NN-GSO), and the classification model. The goal of the hybrid intelligent system is to take the advantages and reduce the disadvantages of the constituent models. The system is capable of learning from data sets and reach great classification performance. Consequently, various algorithms have been developed that include Neural Network based on Galactic Swarm Optimization (NN-GSO), Neural Network based on Particle Swarm Optimization (NN-PSO) and Neural Network based on Genetic Algorithm (NN-GA) to improve NN structure and accuracy rates. For the evaluation process, the hybrid intelligent system has used multiple of benchmark medical data sets to evaluate the effectiveness. These benchmarks were gotten from the UCI Repository of Machine Learning. The three-performance metrics were calculated are accuracy, sensitivity and specificity. These metrics are useful for medical applications. The proposed algorithm was tested on various data sets which represent binary and multi-class medical diseases problems. The proposed algorithm performance analyzed and compared with others using k-fold cross validation. The significance tests results have proven that the proposed algorithm is effective to solve neural networks with good generalization ability and network structure for medical diseases detection.**

Keywords*—Artificial neural network; galactic swarm optimization; particle swarm optimization; genetic algorithm; hybrid intelligent system; medical decision support

I. Introduction

The medical applications consider an important field for researchers. Because healthcare is one of the reasons that will help to encourage the general health and prosperity of the society. The two significant components in medical sciences are prediction and diagnosis of various diseases. In particular, prediction is the disease symptoms and diagnosis is relying on the experience of the physician. Typically, the physician gains the knowledge depending on the symptoms and the diagnosis. However, decision making problems such as prediction and diagnosis include complexity and uncertainty.

The medical knowledge and the treatments are progress very fast such as the appearance of new diseases and drugs. So, the ability of physicians to be aware of all current knowledge and development it is challenging. That's why the deployment of intelligent systems is helpful as medical decision support tool to help physicians in prediction and increasing diagnosis accuracy. Also, it will assist to arrive to decision quickly.

In machine learning, neural networks (NN) are used widely in medical decision support tools and have important advantages for these systems. Artificial neural networks are nonlinear sophisticated modelling techniques inspired by biological nervous systems. Neural networks capture the patterns in data by iteratively adjusting their synaptic weights in line with the learning algorithm [1]. Neural networks are a useful tool for various fields such as classification, prediction, pattern recognition, system identification, signal processing and function approximation. Classification problems consider the most artificial neural networks applications for medicine. There are multiple advantages of neural networks such as avoiding the time wasting and exacting knowledge gain procedure through learning from the data sets the relationship between patient symptoms (input) and the disease (output). However, choosing an appropriate architecture and learning algorithm is very important to have a high efficiency in ANN. In addition to learning, there are other useful properties for neural networks, which involve dealing with missing or incomplete data and filtering noise.

Feed-forward neural networks, in particular Multi-Layer Perceptron's (MLP) has been used in wide range of science and engineering. Because MLP has a high ability to classification and forecasting, it has been widely used in medical diagnosis, detection, and evaluation of medical conditions. It is composed of fully connected feed forward network with one input layer, one or more hidden layers there is a weighted connection between each neuron and all neurons in the next layer. The input layer neurons compute the NN independent variables and output layer neurons will transfer the results.. Between input and output layers there are hidden

layers that can have any number of neurons. In each hidden layer there is defined sequence of activation functions through that the output value will be obtained.

Chitra and Seenivasagam [2] have used a multi-layer feed forward neural network (MLFFNN) and particle swarm optimization (PSO) as a hybrid system for Heart disease prediction at the early stage using the patient medical record. Within specified range this system adopted local and global optimization of the network parameters. Also, Christian and Krzysztof [3] have used feed-forward neural networks for pattern classification with Ant colony optimization (ACO) algorithm as training algorithm. On the other hand, a hybrid Particle Swarm Optimization (PSO) and Gravitational Search Algorithm (GSA) as training algorithm for Feedforward Neural Networks (FNNs) has been done by Seyed Ali Mirjalili et al. [4]. To avoid local minima and enhance the convergence speed. Hamada et al. [5] have used a hybrid system that involves the artificial neural networks (ANN), fuzzy logic, and genetic algorithms (GA). The combination of the neural networks and fuzzy logic helped to improve the performance. The Genetic algorithm have been used to minimize the fuzzy rules and number of features. Moreover, the GA worked on optimizing the initial weights of the artificial neural networks.

This research paper proposes a hybrid intelligent system that consists of artificial neural network (ANN) and the galactic swarm optimization (GSO). This hybrid intelligent capable to learn from data samples to be able to correctly classify the problems. The GSO algorithm is used for NN learning. It is inspired by the motion of stars, galaxies and super clusters of galaxies under the influence of gravity [6]. Comparing to state-of-the-art PSO algorithms the GSO algorithm consider faster in converged to a significantly better solution on a variety of multi-modal and high dimensional benchmark optimization problems. we can conclude from the paper [6] that comparing to other algorithms GSO consider better and gives a good result because of the characteristics that GSO have. On the other hand, a decision support tool with high accuracy, sensitivity, specificity is important for reducing cost and saving time. Furthermore, these metrics are calculated to measure the performance of the classification.

In addition, the aim of the hybrid intelligent system is to incorporate multiple techniques to complement each other and solving each other's limitation. Because the previous meta-heuristics have several disadvantages, the proposed hybrid system uses a new meta-heuristic GSO that use in each epoch in the explorative phase and exploitative phase to prevent premature convergence and allows multi-modal surfaces to be efficiently explored. Also, GSO superior many multi swarm algorithms. On the other hand, FFNN take most of the research interests because of its ability to map any function to an arbitrary degree of accuracy.

The objectives of this paper are to propose a hybrid intelligent system for the design of neural network for medical data classification, to use benchmark medical data sets for evaluating the effectiveness of the system and to evaluate, validate and compare the performance of the proposed hybrid intelligent system with Neural Network based on Particle Swarm Optimization (NN-PSO) and Neural Network based on Genetic algorithm (NN-GA). The rest of the paper is structured as follows. Background and overview of literature review are provided in Section 2. The proposed methodologies are presented in Section 3. Section 4 shows the experimental studies. The results and the significance of the results are shown in Sections 5 and 6, respectively. Section 7 presents the discussion and analysis of the results. The conclusion and future work are presented in Section 8.

II. Background and Literature Review

This section provides a brief explanation of Artificial Neural Network, Genetic Algorithm, Particle Swarm Optimization and Galactic Swarm Optimization along with some of the key basic concepts.

A. Artificial Neural Network (ANN)

An Artificial Neural Network (ANN), also known as a neural network, it is a mathematical model inspired by biological nervous systems that consist of an interconnected group of simulated neurons. Neural networks process information for computation by using connectionist approach and they can model the simple and complex relationships. They are an adaptive system that changes its structure in the learning phase [1].

An ANN can be designed for different type of applications, such as data classification and pattern categorization. Network structures are the arrangement of neurons to form layers and the connection pattern formed within and between layers. There are different types of NN architectures are Multi-Layer Perceptron network, Single-layer Perceptron network, Radial Basis Function network (RBF), Hopfield network and Recurrent network. Multi-Layer Perceptron network (MLP) composed of fully connected feed forward network with one input layer, one or more hidden layers and output layer. There is a weighted connection between each neuron and all neurons in the next layer.

The important aspect of neural network is its capability of learning. The performance of neural network is relying on the success of the training process, and the training algorithm. Training or learning is a procedure of parameter tuning by which a neural network adapts itself to a stimulus and then desired output is produced [23].

B. Genetic Algorithm

A genetic algorithm (GA) can be understood as an "intelligent" probabilistic search algorithm which can be applied to a variety of combinatorial optimization problems. The genetic algorithm originally developed by Holland [25] and it is based on principles of natural evolution [11]. The natural populations develop according to the principles of natural selection and survival of the fittest. The Individuals who have a better chance of surviving and reproducing they are the more successful in adapting to their environment while the less fit individuals fit will be removed. So, in each successive generation, the genes from the highly fit individuals will spread to an increasing number of individuals [7].

Therefore, GA will imitate these procedures by taking an initial population of individuals and using genetic operators to every reproduction. Each individual in the population will be

encoded into a chromosome or string which represents a possible solution to a given problem. The given objective function evaluated the fitness of an individual. The reproducing process will be done to highly fit individuals by replacing parts of their genetic information with other highly fit individuals, in a crossover procedure. Which will result new offspring solutions which share some characteristics taken from both parents. after crossover, the mutation process is applied to individuals by modifying some genes in the strings. Until a satisfactory solution is found, this evaluation selection-reproduction cycle is repeated [7]. The outline of a GA is shown in Fig. 1.

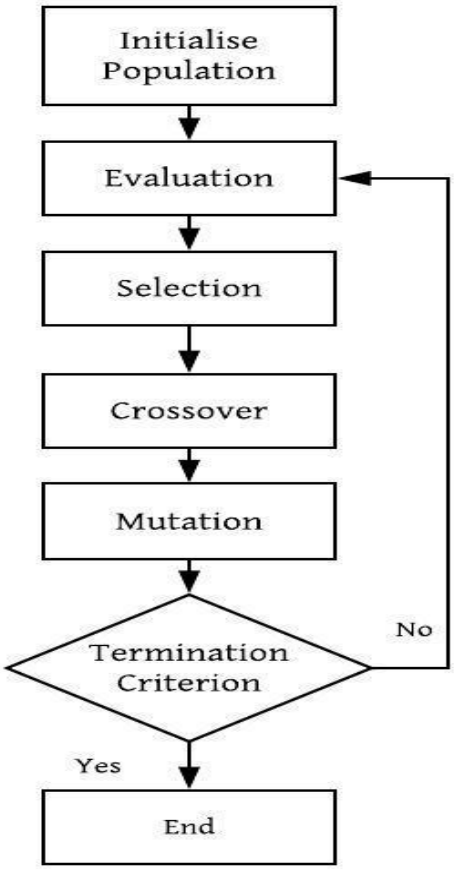

Fig. 1. Outline of GA algorithm [9].

C. Particle Swarm Optimization

Particle Swarm Optimization (PSO) was introduced by Eberhart and Kennedy [24], it is a population-based stochastic optimization technique. PSO doesn't require gradient information of the objective function and with less iteration, PSO can reach the global optimum value [4]. In PSO, each particle in the population has two vectors a velocity vi(t) vector, that enable the particles it to move within the problem space and position xi(t) vector. In the optimization problem, the number of decision variables will identify the dimensions of the two vectors. Updating the particle position is done by using the previous position information and the current velocity of the particle.

$$v_i(t+1) = wv_i(t) + c_1 r_1 (p_i - x_i(t)) + c_2 r_2 \left(p_g - x_i(t)\right) \quad (1)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2)$$

Where v_i is the velocity of particle i , x_i is the position of particle i , c_1 is a weight applied to the cognitive learning portion, c_2 is a similar weight applied to the influence of the social learning portion, r_1 and r_2 are separately generated random number in the range of zero and one. p_i is the previous best location of particle i also known as p_{best}, p_g is the best location found by the entire population, also known as the G_{best}. Fig. 2 shows the basic pseudo-code for the PSO algorithm [8].

```
For each particle do
    initialize particle position and velocity
end for
while stopping criteria are not fulfilled do
    for each particle do
        calculate fitness value using problem specific objective function
        if fitness value is better than best fitness value pbest in particle
            history then
            set current position as pbest
        end if
    end for
    choose as gbest the particle with best fitness value among all particles in
    current iteration
    for each particle do
        calculate particle velocity based on Eq. (1)
        update particle position based on Eq. (2)
    end for
end while
```

Fig. 2. Standard PSO algorithm [8].

D. Galactic Swarm Optimization (GSO)

The GSO algorithm imitates the attraction of stars within galaxy to large masses and galaxies themselves to different large masses [2] as follows: First, according to the PSO algorithm, in every subpopulation the individuals are attracted to better solutions in the subpopulation. Secondly, the best solution for each sub swarm will be treated as a super swarm. The individuals in the super swarm will also move according to the PSO algorithm. The sub swarm is analogous to a galaxy of stars and the super-swarm is analogous to a cluster of galaxies.

In terms of concept and computation, GSO algorithm consider different than other multi-swarm approaches that run independently and continuously a multi-swarm that share information periodically with the slave swarms. While the super swarm in GSO only exists through level 2.

In GSO algorithm, promote exploration and avoid premature convergence consider the main difference between GSO and lots of alternative multi-swarm PSO because the flow of data between the sub swarms and the super swarm is unidirectional. This unidirectional relation means that the super

swarm doesn't influence the exploration of the sub swarms by inserting good solutions just like other multi-swarm approaches. So, the solutions that have been computed by the super swarm not reinserted into the sub-swarms to ensure independent exploration of the sub swarms. Fig. 3 shows the GSO algorithm.

```
Begin PSO Level 1
For each particle
    initialize particle position and velocity
end for
For i=1 to Maximum number of epochs
For i=1 to number of subswarms
  While stopping criteria are not fulfilled do
      For each particle to size of set do
      calculate fitness value using problem specific
                        objective function
          If fitness value is better than the best fitness value pbest in
                        particle history
           then
           set current position as pbest
           end If
      end For
      choose as gbest the particle with best fitness value among all
                    particles in current iteration
      For each particle do
        vi = w1vj + c1r1(pj − xj) + c2r2(gi − xi)

        xi = xi + vi

      end For
  end while
end For
Begin PSO Level 2
For each particle do
initialize particle position with gbest for each sub swarm
end For
        While stopping criteria are not fulfilled do
          For each particle to number of subswarms do
            calculate fitness value using problem specific objective
                                function
              If fitness value is better than best fitness value pbest in
                            particle history then
                    set current position as pbest
              end If
          end For
        choose as global best the gbest with best fitness value among
                    all particles in current iteration
          For each particle do
        v = w2v + c3r3(p − y) + c4r4(g − y)

        y = y + v

          end For
          end While
end For
```

Fig. 3. GSO algorithm [6].

III. METHODOLOGY

A. Neural Network Structure

In this paper, the objective of the hybrid approach is to apply GSO learning algorithm to train the weights of feed forward neural network, a multi-layer perception (MLP) neural network has been used. This network use Three-layers, which the first layer is composed of the input variables, the second layer consists of hidden nodes and the last layer is composed of the output variables [10]. It is built as shown in Fig. 4.

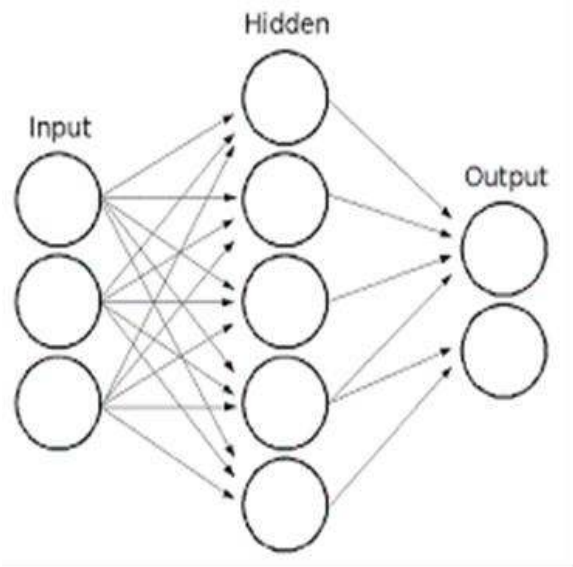

Fig. 4. Structure of neural network [10].

To determine the number of hidden nodes, (3) in [11] has been used as it is shown below:

$$j = \sqrt{n + m + a} \quad (3)$$

Where j is the number of the hidden node, a ∈ [1, 10], n is the number of input nodes and m represents the number of output nodes.

B. Neural Network Representation

NN-GSO and NN-PSO algorithms will be represented by particles, but NN-GA algorithm will be represented by individuals. The representation is formed by four parts: the connection weights between the input layer and the hidden layer w_{ih}, the weights between the hidden layer and the output layer w_{ho}, and the hidden layer bias weights B_h and the output layer bias weights B_o. NN-GSO, NN-PSO and NN-GA algorithms will be encoded as vectors as in Fig. 5 where vectors are sequence of real numbers each of which belongs to the interval [−0.5, 0.5] where the dimension of individuals is given by (4).

$$(i \times h) + (h \times o) + h + o \quad (4)$$

Where i is the number of nodes in the input layer, h is the number of nodes in the hidden layer and o is the number of nodes in the output layer.

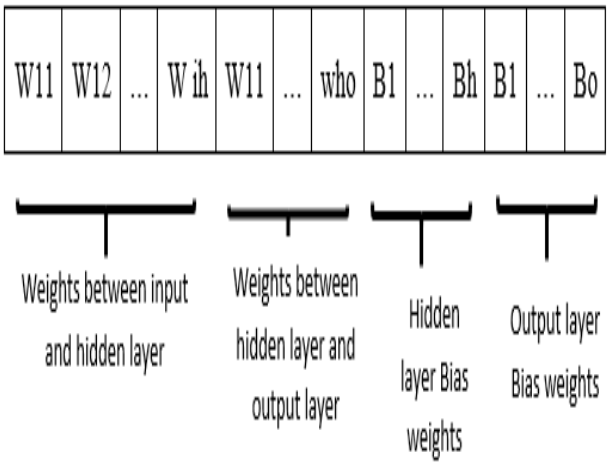

Fig. 5. Representation of NN-GSO, NN-PSO and NN-GA individuals structure.

C. Training Scheme

In this paper, Supervised Learning has been used in which a network defined with a training dataset that contain attributes (input) and output pairs. The learning procedure for the network is done by changing weights at each step of the training in order to reduce the error measure between the network's output and the desired known target value for a given input.

D. Fitness Function

Mean squared error (MSE) is used in this research as performance measure indicators to evaluate the performance of solutions. MSE fitness function considers the most common performance function used to train NN. The weights of the neural network are adjusted to minimize MSE on training set. Equation (5) has been used to calculate MSE.

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\| t_j(i) - y_j(i) \|^2 \quad (5)$$

Where N, $t_j(i)$, $y_j(i)$ are maximum number of patterns, desired outputs and estimated outputs of Neural Network respectively.

E. GSO-Based NN Network Training Algorithm

GSO algorithm has been applied to improve Neural Network in various aspects such as network connection (weights, biases), and learning algorithm. The main process in our research is to apply GSO-based training algorithm on biases and weight optimization and investigate the efficiency of GSO in terms of robustness for training NN and the accuracy rate. By applying GSO algorithm the swarm is subdivided in to sub swarms. The GSO algorithm will update the sub swarms as well as the super swarm using the PSO algorithm.

GSO is a population (swarm) based optimization tool. The swarm is subdivided in to sub swarms and GSO algorithm will update the sub swarms as well as the super swarm according to the PSO algorithm. Every single solution (called a particle) which flies over the solution space in search for the optimal solution. Through following the personal best solution of each particle and the global best amount of the entire sub swarm, the algorithm finishes the first level and the best solution of each sub swarm will participate in the second level the super swarm which also move according to PSO. The particles are evaluated using a fitness function to see how close they are to the optimal solution. The output of this algorithm is weights and biases. The particles (weight, bias) values are initialized randomly. The particles are updated according to (6) and (7):

$$W(t+1) = W(t) + \Delta W(t+1) \quad (6)$$

$$\Delta W(t+1) = w.\Delta W(t) + c_1.rand.[pBest\ (t) - W(t)] + c_2.rand[gBest(t) - W(t)] \quad (7)$$

Where w, c_1, c_2 are inertia, cognitive and social acceleration constant respectively. The flowchart procedure for implementing the GSO is given in Fig. 6.

The Pseudo code for NN-GSO algorithm is as follows:

Step 1: Initialize the network Choose the number of nodes for the input, output and hidden layers

Step 2: Determine the initial value of weights between 0.5 and - 0.5

Step 3: Learning step and calculation of the weight values

```
Define PSO parameters (c1, c2, w, r1, r2)
Initialize population
The swarm will be divided to sub swarms
For each sub swarm
Save best position of any particle (global-best)
Loop
 For each particle in sub swarm
  Compute new particle velocity based on Eq. (6)
  Compute new position based on Eq. (7)
  Compute the error of new position based on Eq. (5)
  If new error better than best-error
   Best-position = new position
  If new error better than global-best
   Global-best = new position
 End for
End loop
Return global-best position
End for
The global-bests will participate on the super swarm population
Save best position of any particle (galactic-best)
Loop
 For each particle in super swarm
  Compute new particle velocity
  Compute new position
  Compute error of new position
  If new error better than best-error
   Best-position = new position
  If new error better than galactic -best
   Galactic-best = new position
 End for
```

End loop
Return galactic-best position
Step 4: give a diagnosis

F. PSO-Based NN Network Training Algorithm

Here we will present the procedure of NN-PSO. PSO is a population (swarm) based optimization tool. Every single solution (called a particle) which flies over the solution space in search for the optimal solution. The particles are evaluated using a fitness function to see how close they are to the optimal solution. The output of this algorithm is weights and biases. The particles (weight, bias) values are initialized randomly. The particles are updated according to (6) and (7).

The Pseudo code for NN-PSO algorithm is as follows:

Step 1: Initialize the network

Choose the number of nodes for the input, output and hidden layers

Step 2: Determine the initial value of weights between 0.5 and - 0.5

Step 3: Learning step and calculation of the weight values

Define PSO parameters (c_1, c_2, w, r_1, r_2)
Loop
For each particle in swarm
Compute new particle velocity based on Eq. (6)
Compute new position based on Eq. (7)
Compute error of new position based on Eq. (5)
If new error better than best-error
Best-position = new position
If new error better than global-best
Global-best = new position
End for
End loop
Return global-best position
Step 4: give a diagnosis

G. GA-Based NN Network Training Algorithm

Here we will present the proposed algorithm of NN-GA, to compare the results of this method with NN-GSO and NN-PSO.

The Pseudo code for NN-GA algorithm is as follows:

Step 1: Initialize the network

Choose the number of nodes for the input, output and hidden layers

Step 2: Determine the initial value of weights between 0.5 and - 0.5

Step 3: Learning step and calculation of the weight values

Creation of the initial population of chromosomes.
For each individual in the population
Loop
Evaluate the fitness of all the chromosomes of the population based on (5).
The best chromosomes will be selected to reproduce, using mutation and crossover.
With the new chromosomes created from the fittest of the previous generation, a new generation is created.
end loop
Evaluate the fitness for all the chromosomes of the population.
Select the fittest chromosome of the population as the new weight.
end for
Step 4: Give a diagnosis

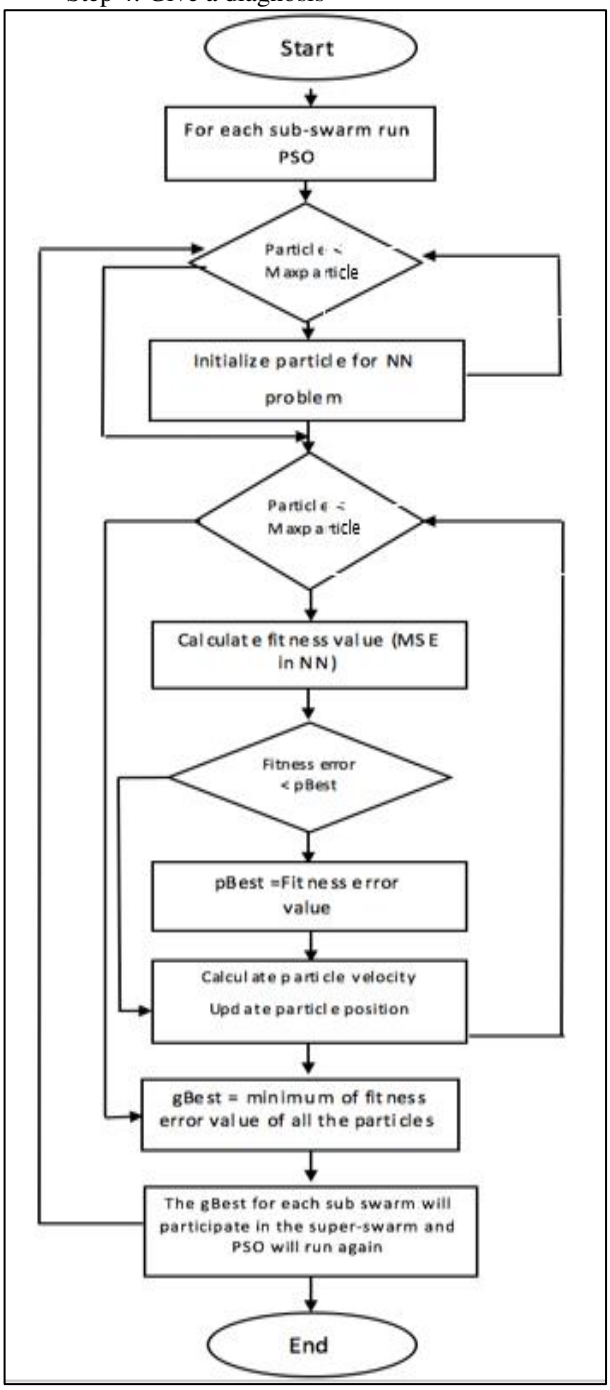

Fig. 6. NN-GSO procedure.

H. Performance Metrics

The performance metrics, accuracy, sensitivity and specificity are used to analyze and compare the outcomes of NN-GSO with NN-PSO and NN-GA algorithms. The accuracy measures how often the classifier makes the correct prediction on a dataset is defined by (8):

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

Where, TP (True Positive): is the proportion of positive cases that are correctly diagnosed as positive; FP (False Positive): is the proportion of negative cases that are wrongly diagnosed as positive; FN (False Negative): is the proportion of positive cases that are wrongly diagnosed as negative; TN (True Negative): is the proportion of negative cases that are correctly diagnosed as negative. On the other hand, sensitivity and specificity are statistical measures of the performance of a binary classification test, also known in statistics as classification function. Sensitivity which is also called the true positive rate, measures the proportion of positives that are identified correctly. Such as the percentage of sick people who are correctly identified as having the condition. This can be expressed mathematically as (9):

$$\text{Sensitivity} = TP / (TP + FN) \quad (9)$$

Specificity which is also called the true negative rate, measures the proportion of negatives that are correctly identified, such as the percentage of healthy people who are identified correctly as not having the condition. This can be expressed mathematically as (10):

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

IV. Experimental Studies

Several experiments were performed on nine real-world medical data sets. The results of NN-GSO for each dataset are compared to NN-PSO and NN-GA algorithms based on their overall classification performance.

A. Description of the Datasets

The proposed approach is applied on nine bench mark data sets for medical diseases for training and testing of the NN-GSO algorithm. These are Breast cancer, Diabetes, Heart, Hepatitis, Liver Disorders and Appendicitis datasets which represent binary class classification problems, while Thyroid, Dermatology and Alzheimer represent multiclass classification problems. All datasets are obtained from the University of California at Irvine UCI machine learning databases [21], except Alzheimer dataset which obtained from Open Access Series of Imaging Studies (OASIS) [22]. Table I shows the different dataset characteristics such as number of patterns and features.

1) Breast cancer dataset

This dataset has been collected from Dr. William H. Wolberg in the Wisconsin Hospitals University, Madison. The main purpose of it is to predict if the patient has benign tumour or malignant tumour. This data set includes 683 examples that have nine inputs (attributes) and two output classes.

2) Diabetes dataset

The dataset purpose is to predict a Pima Indian individual either positive or negative, depend on medical examination and personal data. This dataset considers a difficult problem which contains 768 examples with eight inputs (attributes) and two output classes.

3) Heart dataset

The detection of the existence or absence of heart diseases (heart positive or heart negative) is the main objective of this dataset. There are 303 examples in this dataset of which 139 are positive instances and 164 are negative instances. It has thirteen inputs (attributes) and two output classes.

4) Hepatitis dataset

The objective of this dataset is to detect if a hepatitis patient will die or live. However, this dataset considers noisy and complicated data because it has many missing data. It includes 155 examples that have 19 inputs (attributes) and two output classes.

5) Thyroid dataset

The objective of this dataset is to diagnosis if the patient is normal (1) or suffers from hyperthyroidism (2) or hypothyroidism (3). It contains 2069 examples with twenty-one input (attributes) and three output classes.

6) Alzheimer dataset

This dataset obtained from Open Access Series of Imaging Studies (OASIS) that contains cross-sectional MRI data. The objective of this dataset is to detect if the patient has normal control (NC) or mild cognitive impairment (MCI) or Alzheimer's disease (AD). This dataset consists of 158 subjects aged 30 to 96. It contains fife attributes and three output classes' normal control (NC), mild cognitive impairment (MCI) and Alzheimer's disease (AD).

7) Appendicitis dataset

The objective of this dataset is to predict if the patient has appendicitis (1) or not (0). This data represents seven medical measures (inputs) taken over 106 patients and two output class.

8) Liver dataset

This dataset includes 345 examples to diagnose the existence or absence of liver disorders diseases. It has six inputs and two output classes (abnormal or normal). This dataset has 145 are positive examples (abnormal) and 200 are negative examples (normal).

9) Dermatology dataset

In dermatology, the differential diagnosis of erythemato-squamous diseases considers a real problem because they all share the clinical features of erythema and scaling, with very little differences and in the first stages the disease may show the features of another disease and may have the characteristic features at the following stages. This data contains 34 attributes. The diseases in this data set are psoriasis, seboreic dermatitis, lichen planus.

TABLE I. DESCRIPTION OF DATASETS.

Data Set	Attributes	Classes	Samples
Breast cancer	9	2	683
Diabetes	8	2	768
Heart	13	2	297
Hepatitis	19	2	155
Thyroid	21	3	2069
Alzheimer	5	3	158
Appendicitis	7	2	105
Liver	6	2	343
Dermatology	33	3	242

B. Experimental Setup

The Evaluation of the proposed algorithms has been done by random 10-fold cross validation. In 10-fold cross-validation, the dataset is split in to 10 equal parts .one part used as testing dataset and the other parts used as training dataset. This operation will be repeated until all parts used as testing dataset [9]. We use all datasets to evaluate the performance of proposed algorithms and analyze the evolutionary process of it. Table II shows the parameters settings for the proposed algorithms. While the number of input and output nodes is depending on the problem domain. We use the common parameter settings for GSO, PSO and GA algorithms that are recommended in literature [6], [15]. To unify all algorithms the population size is set for 100 and Maximum number of iterations is set to 1000 for all algorithms.

TABLE II. PARAMETERS SETTINGS FOR THE PROPOSED ALGORITHMS

Neural Network Initialization	
Initial weights	[−0.5,0.5]
NN-GSO	
Maximum number of iterations L1	198
Acceleration coefficient (c_1)	1.4
Acceleration coefficient (c2)	1.4
NN-PSO	
Population size	100
Acceleration coefficient (c_1)	1.4
Acceleration coefficient (c2)	1.4
NN-GA	
Crossover Probability (cp)	0.9
Mutation Probability (mp)	0.5

V. RESULTS

The results of study are presented in this section. Table III shows the Mean Square Error (MSE) of the proposed algorithms NN-GSO, NN-PSO and NN-GA on the training and the testing set on Breast cancer, Heart, Hepatitis, Diabetes, Hepatitis, Liver Disorders, Appendicitis, Thyroid, Dermatology and Alzheimer datasets. Also, the average of MSE for all dataset for each of the proposed algorithms is shown in Table III. The Mean, Min, Max and STDV indicate the mean value, minimum value, maximum value and standard deviation, respectively. Ten-fold cross-validation has been used for all datasets to obtain the results. As can be seen from Table III, the testing error values indicate that NN-GSO algorithm has resulted in better convergence, for most of the data sets compared to NN-PSO and NN-GA algorithms. It also shows that NN-GSO algorithm has produced the smallest error in the testing set for Diabetes, Hepatitis, Breast cancer, Liver Disorders and Dermatology datasets with an average error 0.15650, 0.16568, 0.02729, 0.20814, and 0.16665 respectively. However, NN-PSO algorithm outperforms the others with the smallest error for Heart and Thyroid datasets with an average error 0.08763 and 0.04546 respectively. For Alzheimer and Appendicitis datasets NN-GA algorithm produce the smallest error with an average error 0.03851 and 0.11803 respectively. NN-GSO algorithm does not show good performance for Heart and Thyroid dataset the reason for that is the nature of data. While NN-GA algorithm is better for Alzheimer and Appendicitis datasets because of the small number of samples.

Accuracy, sensitivity and specificity consider the most using performance measures for dataset classification. The measure of the ability of the classifier to obtain an accurate diagnosis is the accuracy. Sensitivity will evaluate the performance of classifier through identifying the positive examples which is the number of false negatives and true positives. While specificity will evaluate the performance of classifier through identifying the negative examples which is the number of false positives and true negatives.

Table IV and Fig. 7 to 9 shows the statistical results for accuracy, sensitivity and specificity of NN-GSO, NN-PSO, and NN-GA algorithms for all datasets on both the training and the testing set. Hepatitis, Thyroid and Liver datasets deserve special mention. These datasets represent difficult classification problems for all algorithms because they are very unbalanced datasets. For Hepatitis and Liver datasets NN-GSO algorithm shows the best results in accuracy and sensitivity, while NN-PSO algorithm shows the best results in specificity. However, NN-PSO algorithm shows the best results for Thyroid dataset in accuracy, sensitivity and specificity with 85.68, 77.03 and 91.98 respectively. As per the results, NN-GSO algorithm outperforms other algorithms in Breast cancer, Diabetes, Appendicitis, Hepatitis, Liver, Alzheimer and Dermatology datasets in accuracy and sensitivity with an average accuracy of 97.09, 77.73, 92.13, 77.33, 70.37, 95.59 and 79.97 respectively. NN-GA algorithm outperforms other algorithms with accuracy 80.95 and sensitivity 80.27 for Heart dataset. As can be seen from the Table IV, NN-GSO algorithm has produced the best results in terms of sensitivity specificity and accuracy, on testing datasets and in average as well. Also, it can be noticed that NN-GSO algorithm has small standard deviation compared to other algorithms.

Overall, NN-GSO algorithm shows better result than other algorithms in Breast cancer, Diabetes, Appendicitis, Hepatitis, Liver, Alzheimer and Dermatology datasets because GSO algorithm uses in each epoch explorative and exploitative phase to prevent premature convergence and allows multi-modal surfaces to be efficiently explored. However, NN-GSO

TABLE III. COMPARISON ON MSE RESULTS OF NN-GSO, NN-PSO AND NN-GA ON THE TRAINING AND TESTING SETS

Data Set		NN-GSO		NN-PSO		NN-GA	
		Training Error	Testing Error	Training Error	Testing Error	Training Error	Testing Error
Breast Cancer	**MEAN**	0.01310	**0.02729**	0.01625	0.03721	0.02283	0.03243
	MIN	0.01290	0.00116	0.01463	0.00146	0.01940	0.00264
	MAX	0.01320	0.06527	0.33498	0.07353	0.02501	0.08697
	STDAV	0.00116	0.02180	0.01461	0.02476	0.00179	0.02524
Diabetes	**MEAN**	0.14838	**0.15650**	0.14473	0.18110	0.17198	0.18864
	MIN	0.14550	0.10363	0.14177	0.14252	0.15823	0.13382
	MAX	0.15410	0.19584	0.22139	0.21819	0.19507	0.24561
	STDAV	0.00280	0.02539	0.00698	0.02256	0.01251	0.03646
Heart	**MEAN**	0.09780	**0.14231**	0.18570	0.08763	0.13060	0.15204
	MIN	0.09520	0.08977	0.12079	0.08017	0.10650	0.09404
	MAX	0.10600	0.24954	0.34086	0.24750	0.15373	0.28738
	STDAV	0.00338	0.04615	0.06862	0.01160	0.01353	0.05803
Hepatitis	**MEAN**	0.08073	**0.16568**	0.05374	0.22422	0.21076	0.25915
	MIN	0.08070	0.08303	0.04286	0.11621	0.19018	0.16649
	MAX	0.08140	0.30362	0.34198	0.47183	0.22937	0.30316
	STDAV	0.00013	0.05891	0.02187	0.10804	0.01407	0.04962
Thyroid	**MEAN**	0.04722	0.05295	0.02036	**0.04546**	0.13991	0.14005
	MIN	0.02630	0.01644	0.01002	0.01032	0.13420	0.10570
	MAX	0.07580	0.08540	0.05374	0.06051	0.14271	0.16201
	STDAV	0.01681	0.02348	0.01292	0.01598	0.00239	0.02186
Alzheimer	**MEAN**	0.02620	0.04633	0.01986	0.06696	0.03765	**0.03851**
	MIN	0.02190	0.00000	0.01537	0.00000	0.02606	0.00439
	MAX	0.03090	0.13514	0.02470	0.18804	0.05996	0.10109
	STDAV	0.00339	0.05291	0.00322	0.06225	0.01013	0.03142
Appendicitis	**MEAN**	0.07323	0.12103	0.02553	0.18110	0.08068	**0.11803**
	MIN	0.05173	0.00861	0.01190	0.14252	0.05474	0.00766
	MAX	0.08887	0.34799	0.08113	0.21819	0.09501	0.36292
	STDAV	0.01276	0.11575	0.02106	0.02256	0.01221	0.12134
Liver	**MEAN**	0.17529	**0.20814**	0.18811	0.21628	0.21842	0.23699
	MIN	0.16796	0.17508	0.18250	0.16903	0.20319	0.21519
	MAX	0.18969	0.23508	0.19688	0.25274	0.23915	0.27375
	STDAV	0.00590	0.02056	0.00461	0.02794	0.01070	0.02411
Dermatology	**MEAN**	0.18748	**0.16665**	0.11720	0.17788	0.25733	0.28699
	MIN	0.16687	0.04269	0.03579	0.03142	0.14592	0.09534
	MAX	0.20753	1.09850	0.15872	1.09239	0.39257	1.18110
	STDAV	0.01128	0.32803	0.03480	0.32330	0.07761	0.31956
Average	**Mean**	0.09438	**0.12076**	0.08572	0.13532	0.14113	0.16143
	STDAV	0.00640	0.07700	0.02097	0.06878	0.01722	0.07640

TABLE IV. COMPARISON OF THE SENSITIVITY (SEN), THE SPECIFICITY (SPE), THE ACCURACY (ACC) OF NN-GSO AND NN-PSO ON THE TRAINING AND TESTING SETS

Data Set		NN-GSO						NN-PSO					
		Training Set			Testing Set			Training Set			Testing Set		
		ACC	SEN	SPE	ACC	SEN	SPE	ACC	SEN	SPE	ACC	SEN	SPE
Breast Cancer	**MEAN**	97.85	97.27	98.93	**97.09**	**96.06**	**98.65**	97.51	96.97	98.51	96.21	95.46	97.37
	MIN	97.40	96.52	98.17	92.96	90.48	95.24	97.22	96.48	97.21	92.65	89.36	90.91
	MAX	98.53	97.99	99.54	100	100	100	97.72	97.51	99.09	100	100	100
	STDAV	0.33	0.41	0.49	2.34	3.51	2.17	0.21	0.37	0.57	2.47	4.00	3.14
Diabetes	**MEAN**	77.73	62.38	85.95	**77.73**	**61.70**	**86.37**	74.31	59.65	82.17	72.41	58.01	80.22
	MIN	76.01	59.17	83.00	71.05	51.72	79.59	70.09	32.65	68.94	64.47	26.09	65.22
	MAX	79.62	65.98	87.97	89.47	78.26	94.34	78.07	78.99	93.07	80.26	76.67	92.45
	STDAV	1.12	2.77	1.56	4.69	8.64	5.25	2.83	12.58	7.43	4.71	16.95	9.44
Heart	**MEAN**	86.83	83.44	89.72	80.33	75.91	**84.91**	90.72	87.65	93.34	77.50	76.34	78.96
	MIN	85.08	80.80	86.71	62.07	50.00	70.59	89.18	82.05	91.03	62.07	41.67	61.54
	MAX	88.81	87.20	93.06	89.66	91.67	92.31	93.28	95.04	96.53	86.21	92.31	92.31
	STDAV	1.37	1.85	1.96	7.99	12.91	6.45	1.40	3.97	1.64	7.16	15.19	9.06
Hepatitis	**MEAN**	88.82	96.68	58.67	**77.33**	**88.44**	33.33	93.26	97.37	77.35	74.67	84.48	39.17
	MIN	86.43	90.99	44.83	60.00	75.00	0.00	90.00	91.89	51.72	46.67	58.33	0.00
	MAX	92.14	100	75.86	86.67	100	66.67	95.71	100	93.10	86.67	100	100
	STDAV	1.73	2.77	8.82	7.17	8.42	23.24	1.93	2.64	10.56	12.88	14.37	33.80
Thyroid	**MEAN**	81.78	69.56	85.38	82.59	69.97	85.61	90.72	84.31	96.35	**85.68**	**77.03**	**91.98**
	MIN	66.45	33.18	72.89	66.18	32.98	74.12	68.87	42.80	93.88	66.51	41.67	83.33
	MAX	90.66	86.53	99.00	93.13	92.48	97.92	98.80	98.04	98.25	99.36	99.49	100
	STDAV	8.28	17.39	8.19	7.83	17.45	7.73	9.88	17.11	1.59	9.89	16.06	6.02
Alzheimer	**MEAN**	97.59	86.60	99.83	**95.59**	**84.17**	**98.52**	98.16	89.29	99.75	92.77	74.17	96.20
	MIN	97.18	82.61	99.16	85.71	50.00	92.31	97.20	85.00	99.16	78.57	0.00	84.62
	MAX	97.90	90.91	100	100	100	100	98.59	91.30	100	100	100	100
	STDAV	0.35	2.37	0.35	4.62	21.68	3.13	0.48	2.53	0.40	7.02	40.91	5.41
Appendicitis	**MEAN**	85.75	60.37	93.77	**92.13**	**63.51**	**99.20**	74.31	59.65	82.17	72.41	58.01	80.22
	MIN	60.00	0.00	77.78	89.58	57.90	97.40	70.09	32.65	68.94	64.47	26.09	65.22
	MAX	100	100	100	94.79	72.22	100	78.07	78.99	93.07	80.26	76.67	92.45
	STDAV	14.24	42.44	8.45	1.60	4.43	0.92	2.83	12.58	7.43	4.71	16.95	9.44
Liver	**MEAN**	73.26	87.44	53.82	**70.37**	**87.62**	46.59	75.39	88.95	56.82	68.96	80.03	**52.65**
	MIN	70.65	84.18	47.69	64.71	78.95	31.25	73.23	83.33	42.86	55.88	57.90	31.25
	MAX	75.49	91.53	60.16	76.47	94.74	70.59	77.74	96.05	61.72	76.47	94.74	76.47
	STDAV	1.70	1.93	3.47	3.73	5.85	11.90	1.45	3.30	5.83	6.38	10.71	15.89
Dermatology	**MEAN**	79.98	72.63	80.10	**79.97**	**78.20**	**80.48**	76.66	85.19	76.62	74.23	73.78	74.29
	MIN	67.15	32.56	66.89	66.67	66.33	66.67	68.63	66.67	68.42	66.67	65.67	66.67
	MAX	97.43	100	98.03	100.00	100.00	100.00	94.99	100	95.18	94.44	95.77	95.83
	STDAV	12.04	26.05	12.20	12.60	12.85	12.85	7.98	17.57	8.07	8.80	9.47	9.13
Average	**Mean**	85.51	79.60	82.91	**83.68**	**78.40**	**79.30**	85.67	83.23	84.79	79.43	75.26	76.78
	STDAV	4.57	10.89	5.05	5.84	10.64	8.18	3.22	8.07	4.84	7.11	16.07	11.26

TABLE IV. (CONTINUED)

Data Set		NN-GA					
		Training Set			Testing Set		
		ACC	SEN	SPE	ACC	SEN	SPE
Breast Cancer	**MEAN**	97.54	97.23	98.15	96.07	95.65	96.53
	MIN	97.24	96.54	97.24	90.14	89.36	90.48
	MAX	98.05	97.99	99.53	100	100	100
	STDAV	0.23	0.44	0.82	3.01	3.79	3.53
Diabetes	**MEAN**	74.67	57.42	83.92	71.37	53.38	80.97
	MIN	69.94	40.57	77.63	59.21	25.00	71.74
	MAX	78.07	65.98	90.31	78.95	69.57	86.96
	STDAV	2.84	7.88	4.61	6.15	13.47	4.66
Heart	**MEAN**	83.94	83.15	84.64	**80.95**	**80.27**	81.88
	MIN	79.85	75.20	77.08	65.52	50.00	62.50
	MAX	87.31	91.74	93.01	89.66	100	100
	STDAV	2.17	5.72	5.56	8.23	13.00	11.39
Hepatitis	**MEAN**	67.74	43.71	87.07	57.00	36.50	**79.11**
	MIN	62.86	13.56	75.34	26.67	0.00	58.33
	MAX	73.57	71.64	98.77	93.33	80.00	100

	STDAV	3.91	18.09	7.56	17.95	22.69	18.40
Thyroid	**MEAN**	64.37	33.55	66.77	64.51	33.81	66.67
	MIN	64.20	33.31	66.67	63.93	33.33	66.67
	MAX	64.78	34.11	67.16	66.79	38.10	66.67
	STDAV	0.19	0.36	0.17	0.88	1.51	0.00
Alzheimer	**MEAN**	95.98	79.21	99.00	95.49	81.67	98.40
	MIN	93.01	59.09	98.32	85.71	50.00	91.67
	MAX	97.89	90.48	100	100	100	100
	STDAV	1.46	8.60	0.66	5.62	24.15	3.38
Appendicitis	**MEAN**	74.31	59.65	82.17	86.75	60.37	80.22
	MIN	70.09	32.65	68.94	60.00	0.00	65.22
	MAX	78.07	78.99	93.07	100	100	92.45
	STDAV	2.83	12.58	7.43	14.14	42.44	9.44
Liver	**MEAN**	66.04	82.57	43.26	59.61	78.67	33.43
	MIN	59.80	71.51	9.30	50.00	64.71	6.25
	MAX	69.68	96.61	60.87	67.65	95.46	60.00
	STDAV	2.99	7.93	16.07	6.23	11.21	16.78
Dermatology	**MEAN**	79.45	54.33	79.71	77.42	74.72	77.86
	MIN	67.34	31.82	66.69	66.67	66.67	66.67
	MAX	91.72	66.67	92.07	86.08	85.64	85.71
	STDAV	8.21	16.87	8.37	7.00	6.63	6.64
Average	**Mean**	78.23	65.65	80.52	76.57	66.12	77.23
	STDAV	2.76	8.72	5.69	7.69	15.43	8.25

Algorithm doesn't show good results for Thyroid and Heart dataset in accuracy, sensitivity and specificity because of the nature of the data is irrelevant, redundant and has huge features.

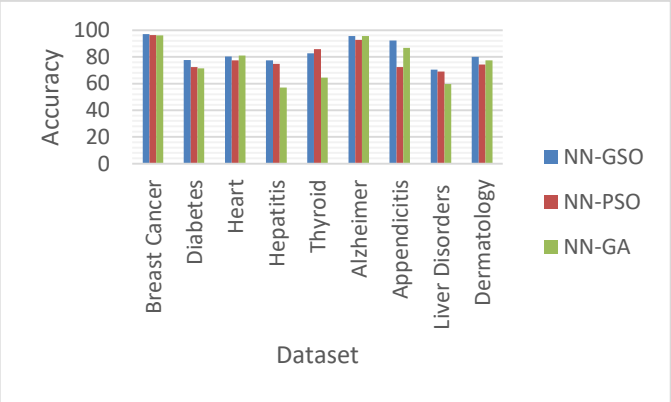

Fig. 7. Comparison of the average accuracy on the testing set.

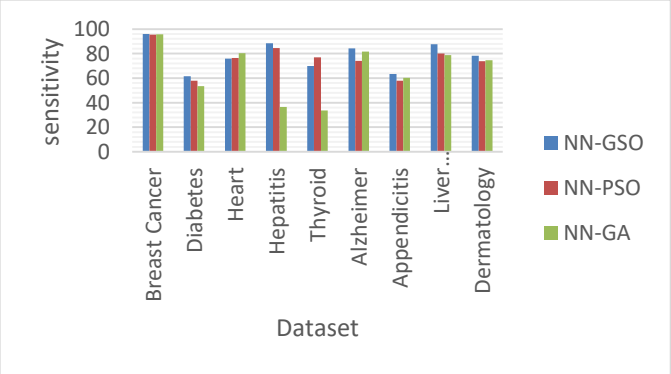

Fig. 8. Comparison of the average sensitivity on the testing set.

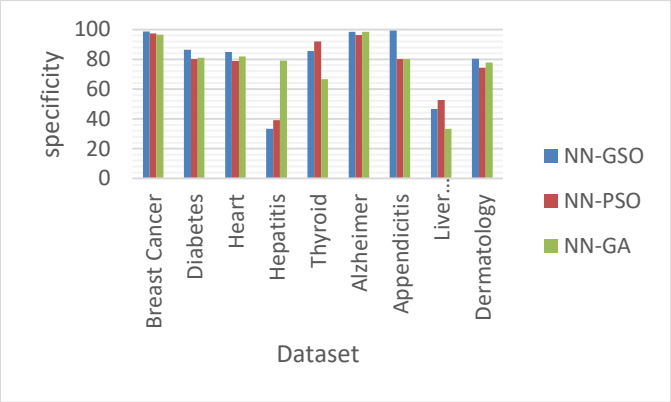

Fig. 9. Comparison of the average specificity on the testing set.

VI. Significance of the Results

As can be seen from the result that NN-GSO algorithm is better in terms of accuracy compared to other algorithms. However, to show that there is a significant difference between the algorithms statistical tests have been applied in this paper. These statistical tests are the Paired t-test test [26] and Wilcoxon's signed-ranks test [27]. They are performed by using 0.05 as a level of confidence (α).

A. Classification Accuracy Rate Analysis

The analysis of the classification accuracy is discussed in this section. Table V presents the average ranks for each algorithm for all datasets. Each algorithm has been ranked by giving the best performance algorithm the rank of 1, the second gets 2 and the third get 3 for each dataset. In case of ties, all tie algorithms get an average rank. Then, we will compare the average ranks between all datasets. As can be seen from Table V, NN-GSO algorithm gets the lower value in the ranking which means that NN-GSO algorithm is the best algorithm. On the other hand, Table VI shows the win, loss and ties count of the number of times that the algorithm is significantly better, loss and tie than other algorithms

respectively. As can be seen from this table, NN-GSO algorithm is statistically different from other algorithms in terms of accuracy rate.

The Wilcoxon's signed-ranks test and Paired t-test has been used to analyze the accuracy for the proposed algorithms to determine whether the proposed methods are statistically different, the null hypothesis states that: H_0: there is no difference between the average accuracy of the proposed algorithms. Table VII shows the results of applying Wilcoxon's signed-ranks test and Paired t-test on the proposed algorithms. As can be seen from Table VII, the values of P-value signed rank and P-value t-test of NN-GSO vs. NN-PSO and NN-GSO vs. NN-GA is lower than significance level, a = 0.05. This means that NN-GSO is statistically different from the NN-PSO and NN-GA. Therefore, the null hypothesis, H_0 is rejected. However, the values of the P-value t-test and P-value signed rank of NN-PSO vs. NN-GA are greater than the significance level, a = 0.05 which means that there are no significant differences between the two algorithms. Thus, the null hypothesis is not rejected. The greater than symbol ('>') and equal sign ('=') mean the algorithm on the left side is significantly and no significantly better than the algorithm on the right side respectively.

TABLE V. RANKINGS OBTAINED FOR THE ALGORITHMS CONSIDERING ACCURACY RATE

Algorithm	Ranking
NN-GSO	1.22
NN-PSO	2.33
NN-GA	2.44

TABLE VI. WINE-LOSS-TIE COUNT OBTAINED FOR THE ALGORITHMS TAKING IN TO ACCOUNT THE ACCURACY RATE

Algorithm	Win	Loss	Tie
NN-GSO	7	2	0
NN-PSO	1	8	0
NN-GA	1	8	0

TABLE VII. PAIRED T-TEST AND WILCOXON'S SIGNED-RANKS TEST RESULTS

Algorithm	P-value t-test	P-value signed rank	Significant
NN-GSO vs NN-PSO	0.0408	0.0195	>
NN-GSO vs GAONN	0.0243	0.0117	>
NN-PSO vs NN-GA	0.5201	0.6523	=

B. Mean Squared Error Analysis

The analysis of the Mean Squared Error (MSE) is discussed in this section. Table VIII presents the average ranks for each algorithm for all datasets. A smaller value in the ranking represents a better algorithm. Table IX shows the win-loss-tie count for the proposed algorithms in terms of average MSE. The results as illustrated in Table IX show that NN-GSO algorithm is significantly better than other algorithms in terms of MSE.

The null hypothesis is stated as, H_0: there is no difference between the average MSE of the proposed algorithms. Table X shows the results of applying both tests (Paired t-test and Wilcoxon's signed-ranks). The P-value t-test and P-value signed rank values of NN-GSO vs NN-PSO and NN-PSO vs NN-GA are greater than significance level a = 0.05 which means there are no significant differences. Therefore, the null hypothesis H_0 is failed to reject. Otherwise, the P-value t-test and P-value signed rank values of NN-GSO vs NN-GA is lower than a, which means that there are significant differences. Therefore, the null hypothesis is rejected.

TABLE VIII. RANKINGS OBTAINED FOR THE ALGORITHMS CONSIDERING MSE

Algorithm	Ranking
GSONN	1.4
PSONN	2.11
GANN	2.4

TABLE IX. WINE-LOSS-TIE COUNT OBTAINED FOR THE ALGORITHMS CONSIDERING MSE

Algorithm	Win	Loss	Tie
GSONN	5	4	0
PSONN	2	7	0
GANN	2	7	0

TABLE X. RESULTS OF PAIRED T-TEST AND WILCOXON'S SIGNED-RANKS TEST TAKING INTO ACCOUNT MSE RATE

Algorithm	P-value t-test	P-value signed rank	Significant
GSONN vs PSONN	0.1204	0.0977	=
GSONN vs GAONN	0.0164	0.0273	>
PSONN vs GANN	0.1009	0.2031	=

VII. DISCUSSION AND ANALYSIS

Based on the above results, this study has been successfully presented a new hybrid intelligent system for the design of neural network for medical data classification (NN-GSO). By using multiple benchmark medical data sets and many performance metrics for evaluating the effectiveness of the system, the classification results show that NN-GSO algorithm has better and acceptable results compared to NN-PSO and NN-GA algorithms. In addition, the computational results of NN-GSO algorithm have been compared with the results of other algorithms in the literature in terms of classification accuracy.

Table XI and Fig. 10 shows a summary of the comparative results. Note that none of the algorithms that presented in Table XI (MEPGANf1-f3 [12], MLP-BP [13], ISO-FLANN [13], NN-CAPSO [14], NN-GSA [14], NN-ICA [14], NN-BP [15], NN-MVO [15], MODE-ESNN [16], DPM [16], SAE-MR

[17], SAE-ZEROMASK [17], MKSVM [18], PMC [19], DG [19], FA [20] and LFA [20]) were tested on all the datasets that had been used in this study. Also, the results presented here are not fine-tuned in any manner, (i.e., the same parameters and experimental settings are used for all datasets). As can be seen from the table, NN-GSO algorithm outperformed other algorithms for breast cancer, diabetes and appendicitis datasets.

However, for heart and liver disorders datasets NN-GSO algorithm provided comparable results to NN-CAPSO [14] and NN-MVO [15] respectively. For hepatitis dataset NN-MVO [15] outperformed other algorithms. PMC [19] algorithm outperformed other algorithms for thyroid and dermatology datasets, but it performed comparable for appendicitis and liver disorders. However, for Alzheimer dataset, NN-GSO algorithm outperformed DPMS [16], SAE-MR [17], SAE-ZEROMASK [17] and MKSVM [18].

Over all, NN-GSO algorithm is better or at least competitive for breast cancer, diabetes, heart, hepatitis, appendicitis and Alzheimer. On the other hand, NN-GSO performs comparable for the liver, disorders, thyroid and dermatology datasets (with respect to classification accuracy) comparing to other algorithms. In short, it has been shown from the results that the NN-GSO is a suitable algorithm that can be applied to solve classification problems because it shows good performance in terms of classification accuracy for most datasets.

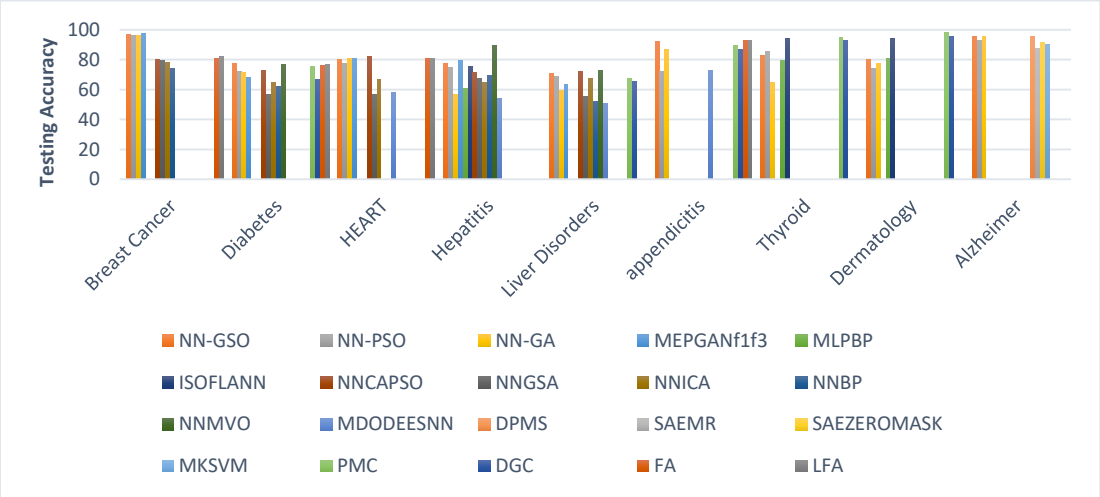

Fig. 10. Performance comparisons of the proposed and existing algorithms on the testing set for all datasets.

TABLE XI. NEURAL NETWORKS CLASSIFICATION ACCURACIES FOR ALL DATASET PROBLEMS WITH CLASSIFICATION ACCURACIES OBTAINED BY OTHER STUDIES

Algorithm / Reference	Breast Cancer	Diabetes	Heart	Hepatitis	Liver	Appendicitis	Thyroid	Dermatology	Alzheimer
NN-GSO	97.09	**77.73**	80.33	77.33	70.73	92.13	82.59	79.97	**95.59**
NN-PSO	96.21	72.41	77.50	74.67	68.96	72.41	85.68	74.23	92.77
NN-GA	96.07	71.37	80.95	57.00	59.61	86.75	64.51	77.42	95.47
MEPGANf1-f3/ [12]	**97.80**	68.35	80.79	79.38	63.50	-	-	-	-
MLP-BP/ [13]	-	-	-	60.83	-	-	79.77	80.63	-
ISO-FLANN/ [13]	-	-	-	75.72	-	-	94.47	94.43	-
NN-CAPSO / [14]	80.25	72.99	**81.85**	71.29	72.32	-	-	-	-
NN-GSA/ [14]	79.25	56.43	56.67	67.74	55.65	-	-	-	-
NN-ICA / [14]	78.25	64.61	66.67	64.52	67.68	-	-	-	-
NN-BP/ [15]	74.41	61.98	-	69.62	52.20	-	-	-	-
NN-MVO/ [15]	-	76.79	-	**89.43**	**72.46**	-	-	-	-
MDODE-ESNN/ [16]	-	-	58.20	54.00	50.57	73.00	-	-	-
DPMS / [16]	-	-	-	-	-	-	-	-	95.35
SAE-MR/ [17]	-	-	-	-	-	-	-	-	87.79
SAE-ZEROMASK / [17]	-	-	-	-	-	-	-	-	91.40
MKSVM [18]	-	-	-	-	-	-	-	-	90.11
PMC/ [19]	-	75.65	-	-	67.25	89.62	**94.67**	**98.09**	-
DGC/ [19]	-	66.62	-	-	65.22	87.13	92.56	95.44	
FA / [20]	80.88	76.04	80.88	-	-	**92.50**	-	-	-
LFA/ [20]	81.94	77.08	80.88	-	-	92.05	-	-	-

VIII. Conclusion and Future Work

In this paper we have proposed a new a hybrid intelligent system as medical decision support tool for medical diseases predication and classification based on the neural network, galactic swarm optimization (NN-GSO). The effectiveness of the proposed algorithm has been evaluated by using multiple of benchmark medical data sets which are Diabetes, Liver Disorders, Heart, Breast cancer, Hepatitis, and Appendicitis datasets which represent binary class classification problems, while Thyroid, Dermatology and Alzheimer represent multiclass classification problems. Experimental results have shown that the proposed algorithm gets better classification compared with NN-PSO and NN-GA algorithms. Multiple of statistical tests have been used to analyze the accuracy of the proposed algorithm. In conclusion, it has been shown that NN-GSO approach is a suitable algorithm that can be employed to solve complex classification problems. For future work, we will focus on improving NN-GSO by training other types of neural networks such as recurrent neural network and implementation of NN-GSO algorithm with different fitness functions. Furthermore, GSO algorithm can be improved for multi-objective algorithm to optimize the structure, number of connections and learning of ANN simultaneously to avoid the problem of trial and error.

Acknowledgment

This work is supported by Al Imam Mohammad Ibn Saud Islamic University (IMSIU) under Research University Grant Scheme. The authors would like to thank Deanery of Academic Research, IMSIU, for the research activities, and Computer Science Department, College of Computer and Information Sciences, IMSIU for the support and incisive comments in making this study a success.

References

[1] Q.Al-Shayea. “Artificial Neural Networks in Medical Diagnosis” IJCSI International Journal of Computer Science Issues, Volume 8, Issue 2, 2011.

[2] R. Chitra and V. Seenivasagam” Risk Prediction of Heart Disease Based on Swarm Optimized Neural Network” Proceedings of International Conference on Computer Science and Information Technology, Advances in Intelligent Syste ms and Computing, 2014.

[3] C.Blum and K.Socha “Training feed-forward neural networks with ant colony optimization An application to pattern classification” Fifth International Conference on Hybrid Intelligent Systems, 2010.

[4] SeyedAli, S. Hashim and H. Sardroudi “Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm” Applied Mathematics and Computation, 2012.

[5] H. R.H.Al-Absi1 , A.Abdullah ,M. I.Hassan, and K. B. Shaban. “Hybrid Intelligent System for Disease Diagnosis Based on Artificial Neural Networks, Fuzzy Logic, and Genetic Algorithms” series Communications in Computer and Information Science, Volume 252, pp 128-139,2011.

[6] V.M.Nakarajan and M.M.Noel. “Galactic Swarm Optimization: A new global optimization metaheuristic inspired by galactic motion” Applied Soft Computing Volume 38, 2016.

[7] Z. Beheshti, S.Hj. Shamsuddin, E.Beheshti and S.S.Yuhaniz.” Enhancement of artificial neural network learning using centripetal accelerated particle swarm optimization for medical diseases diagnosis” Methodologies and Application Soft Computing,2014.

[8] Y.E. Shaoa, C.Houa and C.Chiu. “Hybrid intelligent modeling schemes for heart disease classification” Applied Soft Computing Volume 14, 2014.

[9] S. Qasem, S. Shamsuddin and A. Zain “Multi-objective hybrid evolutionary algorithms for radial basis function neural network design” Knowledge-Based Systems 27, 2012.

[10] Jiake Lv et al “A hybrid approach of neural network with particle swarm optimization for tobacco pests prediction” IFIP International Federation for Information Processing, 2009.

[11] T.Masters “Practical neural network recipes in C++” San Diego [u.a.]: Acad. Press, 1999.

[12] S.Qasem and S.Shamsuddina “Memetic Elitist Pareto Differential Evolution algorithm based Radial Basis Function Networks for classification problems” Applied Soft Computing 11, 2011.

[13] S.Dehuri, R.Roy, S.Cho and A.Ghosh “An improved swarm optimized functional link artificial neural network (ISO-FLANN) for classification” The Journal of Systems and Software 85, 2012.

[14] Z.Beheshti, S. Shamsuddin , E.Beheshti · S.Yuhaniz “Enhancement of artificial neural network learning using centripetal accelerated particle swarm optimization for medical diseases diagnosis” Soft Computing, December 2014.

[15] H.Faris, I.Aljarah, S.Mirjalili “Enhancement of artificial neural network learning using centripetal accelerated particle swarm optimization for medical diseases diagnosis” Applied Intelligence , 2016.

[16] A. Saleh, S. Shamsuddin1and H. Hamed “Multi-Objective Differential Evolution of Evolving Spiking Neural Networks for Classification Problems” IFIP International Federation for Information Processing, 2015.

[17] H. Suk, S. Lee and D. Shen “Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis” NeuroImage, 2014.

[18] S. Liu et al. "Multimodal Neuroimaging Feature Learning for Multiclass Diagnosis of Alzheimer's Disease," in IEEE Transactions on Biomedical Engineering, volume. 62, no. 4, pp. 1132-1140, 2015.

[19] M. A. Mazurowskia , P.A. Habasa, J.M. Zuradaa, J.Y. Lob, J.A. Bakerb and G.D.Tourassib. “Training neural network classifiers for medical decision making The effects of imbalanced datasets on classification performance” Neural Networks 427–436,2008.

[20] M. Alweshah and S. Abdullah “Hybridizing firefly algorithms with a probabilistic neural network for solving classification problems” Applied Soft Computing 35, 2015.

[21] "UCI Machine Learning Repository: Data Sets", Archive.ics.uci.edu, 2018. [Online]. Available: https://archive.ics.uci.edu/ml/datasets.html. [Accessed: 27- Mar- 2018].

[22] "OASIS Brains - Open Access Series of Imaging Studies", Oasis-brains.org, 2018. [Online]. Available: http://www.oasis-brains.org. [Accessed: 27- Mar- 2018].

[23] A. Azzini “A new genetic approach for neural network design and optimization”, Ph.D. dissertation, The University of Milan 2005.

[24] J. Kennedy, R.C. Eberhart, Particle Swarm Optimization, in: Proceedings of IEEE International Conference on Neural Networks, IV, Piscataway, 1995, pp. 1942–1948.

[25] J.H. Holland, Adaptation in Natural and Artificial Systems, University of Michigan Press, 1975.

[26] G. Box, J. Hunter and W. Hunter, Statistics for experimenters. Hoboken, N.J.: Wiley-Interscience, 2005.

[27] F. Wilcoxon “Individual comparisons by ranking methods. Biometrics Bull”,1945.