International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 /doi:10.14445/22315381/IJETT-V69I9P226
Volume 69 Issue 9, 220-229, September, 2021


# A Novel Multi Social Communicative HHO Based Neural Networks with Quasi L1 – Regularization for ASD Bio Marker Identification

N B Arunekumar[1], K Suresh Joseph[2]

[1] *Research scholar, Department of computer science, Pondicherry university, kalapet, Puducherry, India*

[2] *Associate professor, Department of computer science, Pondicherry university, kalapet, Puducherry, India*

[1] arunekumarbala@gmail.com, [2] ksjoseph.csc@gmail.com

***Abstract*** — *The Autism spectrum disorder being a spectrum of the syndrome, may cause impairments in children's cognition capabilities. The study aims to find a biomarker for ASD based on the activation patterns exhibited by the brain image using fMRI. The functional connectivity of various regions in the brain is investigated from the functional image to obtain a pattern that will classify the presence of autism. The functional imaging data were acquired from 17 sites around the globe, totaling 1112 fMRI images in the ABIDE dataset. Based on the HHO algorithm, a Multi Social Communicative HHO is proposed. This algorithm uses sub swarms of HHO with cooperation among the sub swarms. The proposed algorithm is tested over benchmark functions. To elevate the disadvantages of backpropagation in training the ANN over complex datasets such as ABIDE, a NN trained based upon the MSCHHO is proposed. The MSCHHO-ANN is also tested over the Wisconsin breast cancer dataset, which is another generic medical dataset.*

**Keywords —** *Autism Spectrum Disorder, HHO, Artificial Neural networks, Breast Cancer, L1-regularization.*

## I. INTRODUCTION

Autism spectrum disorder is a psychiatric disorder that affects 1% of the population around the globe [1]. The children affected with autism may exhibit a wide spectrum of characteristics such as lack of capability in socializing, repetition in behavior or repetition in words, impulsivity, and even self-harm may occur [2]. Identifying the disorder and giving personal care at the early stage will help the children in improving their social abilities. The pediatrician plays a vital role in identifying autism disorders in children. But the disorder having various symptoms makes it hard to diagnose conclusively in all the affected cases at early stages. In order to find a conclusive biomarker, the presented work proposes a robust Neural Network(NN), which is trained based upon the newly proposed Multi Social Communicative Harris Hawk Optimization (MSCHHO) algorithm.

Artificial neural networks (ANN) are a well-renowned machine learning algorithm that can classify non-linearly separable multi-class data. The universal approximation theorem for ANN states that any complex continuous function can be approximated with one hidden layer. The ANN needs a large sample of data for training to classify the data with the least error. Most of the variants of ANN use gradient-based algorithms for parameter optimization using the learning rate. These learning algorithms may tend to overfit the given dataset such that the training error becomes zero. To avoid such situations, overfitting methodologies like regularization are being used. The regularization reduces the complexity of the model by restricting the weights within a range. Regularization can also be used as an embedded feature selection model.

The L1 regularization method simultaneously acts as a features selection model along with restricting the weights. This paves the way for the learning algorithm to get stuck in sub-optimal solutions or converge very slowly towards the best optimal solution. In recent years, several studies have attempted to solve this problem by solving the parameters of the ANN using meta-heuristic optimization algorithms. The weights and the biases, which are the core components of the ANNs are being determined by meta-heuristic algorithms. The multi swarm behavior is the process of using many sub swarms rather than using a single population. The entire population, instead of following a single best solution or single target they are split into subpopulations. Each sub-population will search based on its swarm's target. The multi swarm behavior will help to increase the exploration capabilities and also perform parallel searches over the area.

In this study, a multi swarm cooperative HHO [3] is proposed by implementing multiple swarms of HHO, which socialize and help to communicate with the weakest swarm using information sharing. The L1 regularization uses gradients that need to be backpropagated for feature selection during the training. The MSCHHO-NN with Quasi L1 regularization is proposed based on the MSCHHO algorithm. With the Metaheuristic optimization of parameters, there is no need to backpropagate through all gradients, and therefore an efficient Quasi L1 regularization for embedded feature selection is proposed along with the MSCHHO-NN.

## II. RELATED WORK

Symbiosis-based alternative learning multi Particle Swarm Optimization (PSO) [4] uses a learning method to choose between four variants of information exchange between and inside swarms. Over the pbest, S-gbest (sub-swarm gbest), gbest, and swarm center, and three are used interchangeably for each variant. The authors have divided the swarm into three parts where each sub-population is updated based on PSO or local search operator with 40% probability, and the elite solutions in each sub swarms learn from the elite solution of other sub swarms. The proposed algorithm is tested over the wrapper Feature selection problem for KNN. The improved multi swarm PSO [5] uses population exchange between swarms by increasing the information communication linearly and linearly decreasing the boundary-based random mutation. The self-adaptive multi-population-based Jaya algorithm[6] for iterating the population is split into sub-populations. The updated equations with best and worst solutions are carried out, and the population is again merged together. A multi-population with 2 sub swarms of cooperation and competitive strategy is introduced in work [7]. Recombination is introduced for migration to completive state from cooperative state and three other strategies such as unique encoded number (UEN) to trace individuals, auto-population dropout (APD) to eliminate non-contributing individuals, and variable migration to induce combination and mutation between two populations. The multi-population Differential Evolution (ED) [8] splits the population into three groups based on fitness. A novel random mutation strategy is proposed and utilized on evolution. After completion of iteration, regrouping of the population is done regularly, and doping where random individuals are spawned to replace the worst individuals is carried out when the current iteration is above two-thirds of maximum iteration. The Whale Optimization Algorithm (WOA) population is initialized using chaotic maps into multiple Swarms clustered by the location using k-Means algorithm and each agent is updated based on the best solution of sub population. The Swarm is re-initialized after a threshold of iterations and the proposed algorithm is tested over 9 benchmark functions. The parameter optimization of RBF-SVM was carried out for Wisconsin dataset where accuracy of 96.8%, sensitivity of 97.07% and specificity of 96.3% has been obtained. Based on Resting-State Functional Magnetic Resonance Imaging Data Using Convolutional Neural network the authors have proposed a mixture of convolution neural network experts (MCNNE), which is a gated model that splits the workspace across the NN experts. The pre-processing such as slice time correction, motion correction, functional, structural co-registration, spatial normalization, smoothing with Gaussian kernel, and numerical normalization were carried out. The convolutional Neural Network (CNN) used in this proposed model is trained using transfer learning from a pertained model. The proposed architecture is trained over adamax optimizer on ABIDE I and ABIDE II datasets with an accuracy of 0.7, sensitivity 0.59 and specificity 0.79. In this paper [9]. The authors have utilized an autoencoder for dimensionality reduction and a deep neural network for the classification of the data set. However, the mean accuracy of 0.7 with a sensitivity of 0.73 and specificity of 0.63 is achieved for multisite data. A complete deep learning pipeline is used for the classification of 4D fMRI in this paper [10]. Initially, 3D CNN is used for spatial dimensionality reduction followed by the 3D C-LSTM for feature learning in the spatiotemporal domain and used 3DCNN for further reduction in feature size and finally fully connected layer for classification. In this work, only two sites were considered for classification, and an accuracy of 0.77 and 0.71 was obtained individually on each site. The Metaheuristic algorithms are well renowned for their efficacy in solving nonlinear problems. These swarm intelligence algorithms are one among the subclass of bio-inspired algorithms that mimic the swarming behavior of the species around the globe. Some of the major Swarm Intelligence(SI) algorithms are the PSO [11], which uses the movement of particles towards the global best using the personal best. The Grey Wolf Optimizer (GWO) [12] is one of the efficient and highly used algorithms for solving engineering problems that works based on the encircling and hunting behavior of grey wolves. The WOA uses the bubble net feeding [13] strategy of the humpback whales to hunt the fish. The bat [14] algorithm utilizes the echolocation behavior of the bat to search the prey to hunt it down. The other algorithms which are primly utilized over several engineering problems are the Cuckoo Search Algorithm [15], Firefly algorithm [16], Moth flame optimization algorithm, and salp swarm algorithm [17]. The other famous evolutionary and mathematical-based algorithms are the Differential evolution [18] which evolves a solution based on the other solutions in the population, and the sine cosine [19] optimization algorithm, which is mathematical-based optimization which utilizes the sine and cosine functions to find the optimal solution.

## III. METHODS

### A. HHO- the Harris Hawk optimization algorithm

The Harris Hawk optimization (HHO) Algorithm is one of the recently proposed optimization algorithms that use the strategy of Harris hawks that pounce on the rabbit to hunt them. The rabbits, on the other side, also run to escape themselves from the hunter. This chase of life and death is modeled in the HHO algorithm for optimization. The HHO algorithm is comprised of two different stages, namely exploration and exploitation.

#### a) Exploration

The exploration has two different mechanisms based on the probability value p as in the Eqn. (1). If p is less than 0.5, the positions are updated using the random Hawk, and alternatively, when p is greater than or equal to 5, the solution is updated based on the rabbit's value. The average of all solutions is given in Eqn. (2). The set of variables "a" are random values between range 0 and 1. Equation (3) controls the HHO to be guided into exploration or exploitation.

$$\mathrm{X}(t+1)=\begin{cases} \mathrm{X}_r(t)-a_1\left|X_r(t)-2a_2X(t)\right| & p<0.5 \\ X_{rabbit}-X_{avg}(t)-a_3(LB+a_4(UB-LB)) & p\geq 0.5 \end{cases} \qquad (1)$$

$$X_{mean}(t) = 1/N \sum_{i=1}^{N} X_n(i) \quad (2)$$

$$E = 2E_{\alpha}(1 - i/I) \quad (3)$$

***b) Exploration phase***

The exploitation phase in the HHO algorithm uses 4 different strategies to fine-tune and search for the best solution. The exploitation phase can be categorized into soft and hard besiege operations with normal and rapid dives

**Soft besiege:** The soft besiege phase is the process where the hawks update themselves depending on the position of the rabbit and its jumping energy, respectively, as in Eqn. (4). The jumping energy in Eqn. (5) is based on the position of the rabbit and the current hawk location.

$$X(t+1) = dX(t) - E^* \mid j * X_{rabbit}(t) - X(t) \mid \quad (4)$$

$$dX(t) = X_{rabbit}(t) - X(t) \quad (5)$$

$$J = 2(1 - a_5) \quad (6)$$

**Hard besiege:** The hard besiege directs the hawks towards the position of the rabbit.

$$X(t+1) = X_{rabbit} - E^* \mid dX(t) \mid \quad (7)$$

**Soft besiege and progressive dives:** In this strategy, the hawks adopt either a dive analogous to soft besiege as in Eqn (8) or based on Levy Flight as in the Eqn(10). If the new solution is better than the current solution, then it is adapted as in Eqn (12).

$$G = X_{rabbit}(t) - E^* \mid J * X_{rabbit}(t) - X(t) \mid \quad (8)$$

$$H = G + S * LF(D) \quad (9)$$

$$LF(x) = \frac{(u * \sigma)}{\mid v \mid^{(1/\beta)}} \quad (10)$$

$$\sigma = \left( \frac{\Gamma(1+\beta) * \sin(\frac{\Pi\beta}{2})}{\Gamma\left(\frac{1+\beta}{2}\right) * \beta * 2^{(\beta-1)/2}} \right) \quad (11)$$

$$X(i+1) = \begin{cases} G \text{ if } F(G) < F(X(i)) \\ H \text{ if } F(H) < F(X(i)) \end{cases} \quad (12)$$

**Hard besiege and progressive dives:** In this strategy, the hawks adopt the dives analogs to hard besiege or based on the levy flight as in Eqn. (15)

$$G' = X_{rabbit} - E \mid J * X_{rabbit}(i) - X_{mean}(t) \mid \quad (13)$$

$$H' = G' + S * LF(D) \quad (14)$$

$$X(i+1) = \begin{cases} (G' \text{ if } F(G') < F(X(I)) \\ H' \text{ if } F(H') < F(X(I))) \end{cases} \quad (15)$$

### *B. Artificial Neural networks*

Artificial neural networks are a class of machine learning algorithms that is capable of approximating non-linear functions. The intuition of ANN is to reflect the functionality of neuronal activity of the brain. The structure of the NN comprises 3 main components, which are an input layer, one or more hidden layers, and an output layer shown in

Figure **1**

A feed-forward network that has no feedback connections can be given as $y = f(x, \theta)$. Where x is the input and $\theta$ is the set of learnable parameters that include weights and biases. The MSE function is used as the cost function to find out the error between the predicted results and the actual results, as given in Equation

( 16 ). The value $f^*(x_i)$ is the result that has to be predicted and $j(\theta)$ is the loss that has to be minimized.

$$J(\theta) = 1/N * \sum_{i=1}^{n} f^*(x_i) - f(x; \theta))^2 \quad (16)$$

The output of the 1st and 2nd hidden layer with activation function is given in Eqn. ( 17 ). The output of the 1st layer is given as the input to the 2nd layer.

$$\begin{aligned} h^{(1)} &= g^{(1)}(W^{(1)T} * x + b^{(1)}) \\ h^{(2)} &= g^{(2)}(W^{(2)T} * h^{(1)} + b^{(2)}) \end{aligned} \quad (17)$$

The non-linear activation functions are introduced in the hidden layer to aid the ANN in approximating non-linear functions. The g(x) is a non-linear function that transforms the output value of the hidden layers. Some of the primarily used activation functions are *sigmoid, tanh, and Rectified linear units (Relu)*. The Relu function outputs only positive domain values and equates negative values to zero-g(z)=max{0,z}.

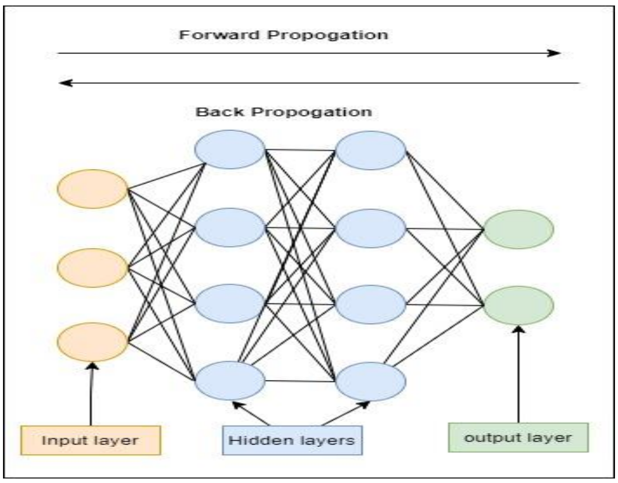


**Figure 1 - Artificial neural networks**

$$f(x) = 1(x < 0)(\mu x) + 1(x \geq 0)(x) \qquad (18)$$

The “dying Relu” problem occurs due to the zero value for the negative domain, and this problem is addressed by the Leaky Relu as denoted in the Eqn ( 18 ) in which μ is a small positive slope of 0.01. Instead of zero as given in Relu for the negative domain, a small positive value of 0.01 is given.

### *C. Regularization*

Regularization is a process of reducing overfitting in machine learning algorithms. The regularization weight is added along with the error function of the learning algorithm, as given in equation (19 )

$$\sum_{i=1}^{n}(y_i - \sum_{j=1}^{m} x_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{m}|\beta_j| \qquad (19)$$

The coefficient $\lambda$ is a hyper parameter that controls the importance of regularization. The higher the regularization lower will be the complexity of the model. The primarily used two types of regularization are L1 and L2 regularization. In L1 regularization, the absolute sum of all weights in the neural network is added to the loss function. This L1 regularization also acts as an embedded feature selection algorithm that regulates the gradient all along with the backpropagation such that more weights are assigned to important features.

### *D. Autism Spectrum Disorder*

The commonly used methodologies for screening of autism is the questionnaire where the expert would test the children with the set of questions based on which the expert decides the possibility for the children to be affected by the disease. As the disease entirely relies on the cognition capacity or the functional behavior of the brain, this work proposes a methodology to identify the ASD based on the neuronal activity of the brain.

The neurons are the functional units that give us cognition capacity. The brain consumes oxygen in the parts where the process of cognition happens. The bold signals are being recorded using this phenomenon of consuming oxygen by the brain. The action potential is created in a neuron when sufficient potentials are being acquired by its dendrites from the other neurons. Once the threshold is attained, the potential flows through the action towards the terminal button. These potentials are being recorded by the fMRI imaging modality. To identify the map of a brain for a respective activity, the bold signal peaks with respect to doing any activity will be recorded. In this case of diagnosing disease, a special fMRI called the resting-state fMRI is recorded. The subjects are made to rest without perceiving any activity for acquiring the rs-fMRI. By finding the correlation of signals for a process, the functional connectivity matrix between the regions of interest in the brain can be built.

The classification can be done based on this connectivity Matrix to identify a biomarker for ASD. In this study, the connectivity matrix, which is based on the CC200 atlas, is being extracted using the Karl Pearson correlation. The C-PAC pre-processing pipeline was utilized by the pre-processed ABIDE dataset before registration of the brain over CC200 atlas.

The ABIDE dataset totally had 1112 samples, where 539 are ASD and 573 are control types. The dataset had samples from 17 international sites whose ages where ranging from a minimum of 7 to a maximum of 64.

## IV. PROPOSED WORK

### *A. Abide pre-processing data pipeline*

The pre-processed ABIDE data set which uses the C-PAC pipeline is considered for this experiment. The C-PAC pipeline uses two types of pre-processing which are structural and functional methods.

#### *a) Structural pre-processing*

The Structural pre-processing will eliminate noise in the brain structure and align various parts of the brain using the methods such as skull stripping, bandpass filtering, and registration. The skull stripping helps to remove the skull from the fMRI. The information about the skull is not necessary for the current experiment. After stripping, different parts of the brain are segmented according to the categorization of tissue type using the FSL tool. Once the segmentation is over, the image is normalized in order to register over the atlas.

#### *b) Functional Pre-processing*

Functional pre-processing is utilized to remove all the unwanted features that are present at the time of recording the image. The steps in the functional pre-processing include slice time correction, motion correction, mean intensity normalization, and signal regression. The slice time correction is a vital pre-processing step that helps to get a complete image over the process of scanning that happens in an interleaved manner in the scanner. The next main artefact is seen in all the images recorded is the minute motion of the head. However, the process of scanning is being controlled. Still, the minute motion will prevail. In order to correct it, the motion correction with respect to a fixed position is carried out. For the purpose of computation, the wide range of intensity recorded is normalized.

#### *c) Feature Selection*

The pre-processed dataset consists of a one-dimensional bold signal for each region. From this BOLD signal, the partial correlation coefficient is being calculated between each pair of regions. The correlation matrix is flattened to form a vector of one dimension, which is used as the sample input. Along with correlation information, the phenotypic information of each sample is also used for training the algorithm. Early to the classification using MSCHHO-NN, the features are reduced using the isolation forest feature importance selection method. With the importance threshold being fixed at 0.0001, a feature vector of around a size of 2300 is obtained.

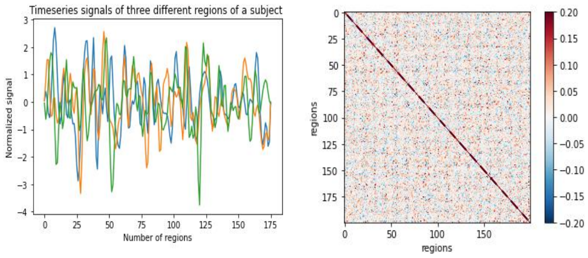


**Figure 2 - BOLD signal and Connectivity matrix from a sample of NYU site**

### *B. Multi Social Communicative HHO*

The population is being divided into several subpopulations into multi Swarm strategies, thus ensuring a diversified search for the optimal solution. In dynamic swarm methodology, the initial number of diverse swarms is being joined into a single social homogeneous Swarm. The initial number of swarms is being decided according to the computational capacity, which may be 5-7 in general as tested with the proposed algorithm, according to the factor, where high diversity must be maintained in the exploration phase, and more local search must be established in the exploitation phase. The numbers of swarms are being decreased such that all the particles perform the local search in the exploitation phase. The numbers of swarms are being decreased based on the variable dynamic Swarm (DS), which decreases in a linear fashion.

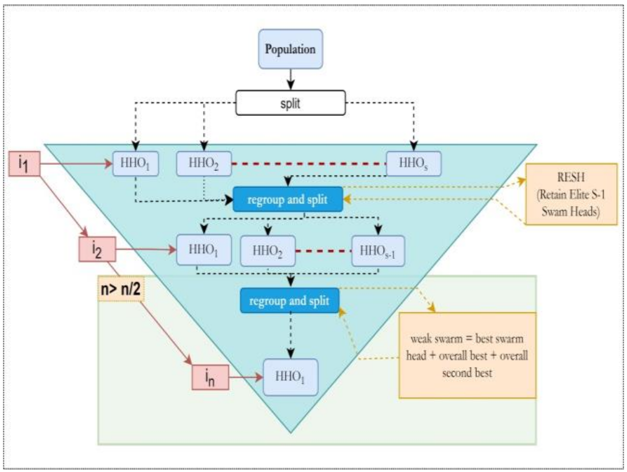


**Figure 3 - Multi Social Communicative HHO**

For each Swarm, an approximately equal number of individuals are being allocated in a round-robin manner. If the dynamic swarm count is lesser than the existing Swarm count, then one among the swarms will be dropped to match the DS count and the existing swarm count. In order to make the swarm exhibit cooperation, once the condition of eliminating a Swarm is encountered, the entire population is merged and again reallocated to the new Swarm groups.

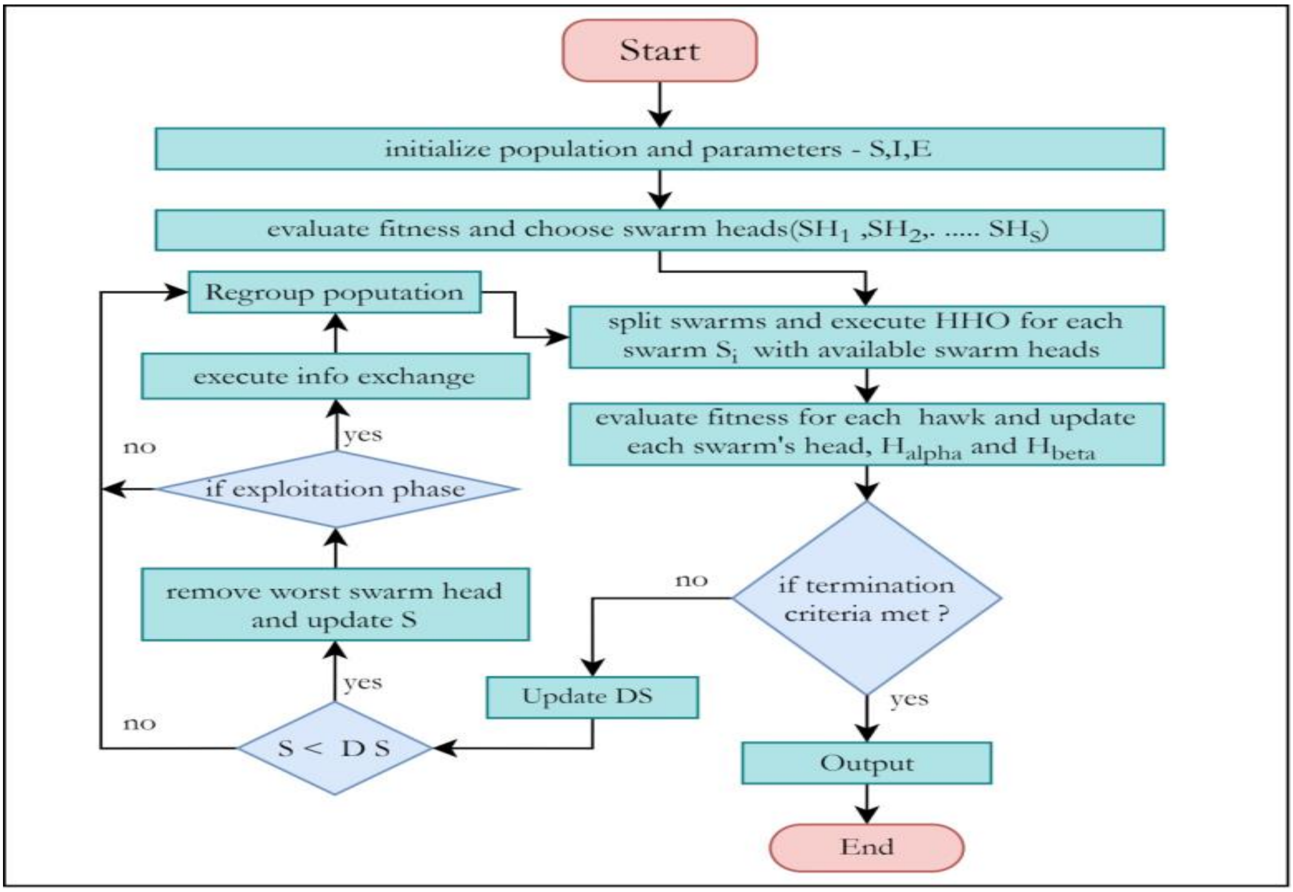


**Figure 4 - MSCHHO flow diagram**

In such a process of reallocation, elitism is maintained. The worst solution of the Swarms is eliminated, and the best fit solution of most elite swarms is retained.

The swarm count (S) is the initial number of swarms. As the iterations go on, the swarm count will decrease to make the population homogeneous. At the final iteration, the entire population will perform exploitation against the best solution. The dynamic swarm count is given by,

$$DS = \lfloor tswarms * ((I - i)/I) \rfloor \quad (20)$$

If the dynamic swarm count is lesser than the existing count, the entire population is merged and reallocated under the existing swarm heads. Then, an HHO update inside each swarm is carried out. The modified exploration phase at each sub swarm is given by Eqn. ( 21 )

$$X(t+1)\begin{cases} X_r^s(I) - a_1 | X_r^s(t) - 2a_2 X(t) | & d < 0.5 \\ X_r^s - X_{mean}^s - a_3 & d \geq 0.5 \end{cases} \quad (21)$$

The average of each sub swarm is given in the Eqn ( 22 ). The exploitation phase of each sub swarm is given in the equations ( 23 ) to ( 27 ), where the update action is based on the target solution of each sub swarm based on the learning rate. In order to overcome these difficulties, a neural network algorithm whose weights and biases are trained using the proposed MSCHHO algorithm is proposed.

$$X_{avg}^S(i) = 1/N \sum_{i=1}^{N} X_n(i) \quad (22)$$

$$X(i+1) = dX(i) - E \,|\, j * X_{rabbit}^s - X(i) \,| \quad (23)$$

$$dX(i) = X_{rabbit}^S - X(i) \quad (24)$$

$$G = X_{rabbit}^S - E^* \,|\, j * X_{rabbit}^S(i) - X(i) \,| \quad (25)$$

$$G = X_{rabbit}^S - E^* \,|\, j * X_{rabbit}^s(i) - X(i) \,| \quad (26)$$

$$G' = X_{rabbit}^S - E \,|\, J * X_{rab}^S(i) - X_{avg}^S(i) \,| \quad (27)$$

In the exploitation state, in order to increase the flow of information among the sub swarms, information exchange is enabled. In the information exchange phase, the weakest swarm head is chosen, and it's strengthened with the overall best solution, the second-best solution, and the one itself. By averaging all three solutions, a location to the center of these three points is chosen, and the weak swarm is directed towards it.

### *C. MSCHHO-NN*

The ordinary NN uses gradient-based learning algorithms, which are prone to get stuck in the local optimum and exhibit slow convergence behavior.

The proposed ANN has the architecture of one input layer, one hidden layer, and an output layer. The size of the input layer is equivalent to the number of features of the dataset. The size of the hidden layer is fixed by the user, and the size of the output layer is equivalent to the number of classes in the dataset. The Proposed ANN uses leaky Relu activation function as nonlinearity and MSE as the cost function. The total dimension that has to be optimized by MSCHHO includes the weights and biases present in the network. The weights and biases are encoded as given in Figure 4, which starts from the weights of the input layer to 1st hidden neuron till the weight of the last hidden layer neuron to the output layer and follows by the weights are the biases of the hidden layer.

| | Weights | | | Biases | | |
|---|---|---|---|---|---|---|
| $\overline{X}_t$ | $w_{i,2:1,1}$ | … | $w_{h,2:1,1}$ | $b_{h,1}$ | … | $b_{h,m}$ |

**Figure 5 Encoding of weights and biases**

### *D. Quasi-L1 regularization*

$$\sum_{i=n}^{n}\left( y_i - \sum_{j=1}^{m} x_{ij}\beta_j \right) + \lambda \sum_{j=1}^{m} | \beta_j | \quad (28)$$

The MSCHHO-NN utilizes a non-gradient algorithm by which there is no need of providing information about the entire network. Thus restricting all the weights to choose a channel that back propagates the gradients for feature selection in the initial layer need not be used. Instead, it is enough to provide the information of weights between the input and the first hidden layer and let the network choose the exact weights that are needed.

Thus only the information of 1st layer is provided with the regularization term as in Equation ( 28 ). The final objective for the optimization of MSCHHO-NN with Quasi L1 (QL1) regularization is a minimization function and given in the Eqn. ( 29 )

Minimize

$$f(\sum_{i=1}^{n} (y_i - \sum_{j=1}^{m} x_{ij}\beta)^2 + \sum_{j=1}^{m} | \beta |) \quad (29)$$

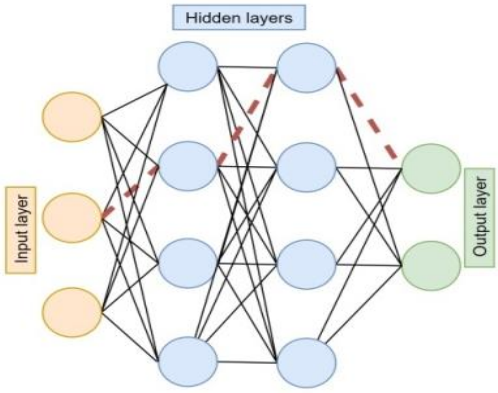


Figure 6. a

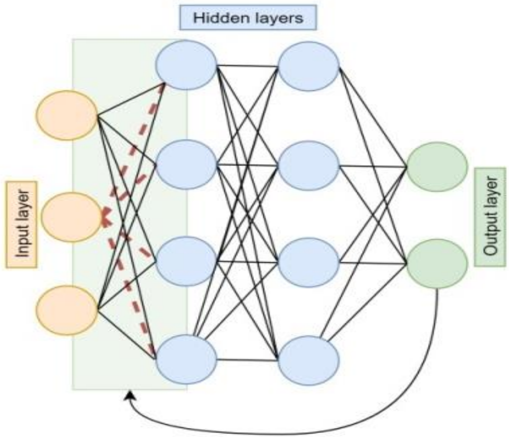


Figure 6. b

**Figure 6 Ordinary L1 vs. Quasi L1 regularization**

**Table 1 Fitness comparison over benchmark functions**

| metrics | Fn. No. | MSC-HHO | GWO | PSO | WOA | CS | SCA | BAT | HHO | JAYA | MFO | FFA | DE | SSA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mean | F1 | **1.94E-15** | 85748.17 | 169018.3 | 0.082937 | 370092.7 | 570371.5 | 781118.9 | 1.08E-11 | 439946.7 | 1400422 | 1088380 | 982693.1 | 90671.25 |
| Stddev | | **8.66E-15** | 10084.86 | 15587 | 0.215766 | 46197 | 154597.3 | 228243.7 | 4.62E-11 | 54501.52 | 37285.78 | 62163.49 | 99707.15 | 6293.61 |
| Rank | | 1 | 4 | 6 | 3 | 7 | 9 | 10 | 2 | 8 | 13 | 12 | 11 | 5 |
| Mean | F2 | **3.83E-08** | 379.8554 | 3.69E+164 | 0.006983 | 2.16E+102 | 203.2432 | 4.73E+274 | 2.69E-06 | 5732160 | 6.03E+235 | 6.10E+190 | 1804.415 | 525.1765 |
| Stddev | | **1.06E-07** | 27.11038 | inf | 0.009616 | 1.16E+103 | 141.2541 | inf | 1.26E-05 | 30862372 | inf | inf | 105.1911 | 19.27854 |
| Rank | | 1 | 5 | 10 | 3 | 9 | 4 | 13 | 2 | 8 | 12 | 11 | 7 | 6 |
| Mean | F3 | **0.384669** | 3083844 | 3366260 | 52794269 | 4782830 | 14639014 | 48123603 | 422603.1 | 21787898 | 7825997 | 10771605 | 9452815 | 1367576 |
| Stddev | | **1.992408** | 416209.8 | 1397140 | 28634016 | 1202033 | 3536023 | 30032093 | 1290912 | 8920996 | 1275351 | 3642526 | 2286922 | 619334.5 |
| Rank | | 1 | 4 | 5 | 13 | 6 | 10 | 12 | 2 | 11 | 7 | 9 | 8 | 3 |
| Mean | F4 | **1.17E-09** | 80.11019 | 52.96172 | 92.09084 | 73.96663 | 99.45167 | 93.15027 | 2.57E-08 | 99.52642 | 98.75257 | 97.64241 | 99.48368 | 34.95552 |
| Stddev | | **2.64E-09** | 3.207191 | 2.903809 | 9.978652 | 5.850737 | 0.297458 | 4.118219 | 9.74E-08 | 0.225654 | 0.366546 | 0.779235 | 0.242024 | 2.270116 |
| Rank | | 1 | 6 | 4 | 7 | 5 | 11 | 8 | 2 | 13 | 10 | 9 | 12 | 3 |
| Mean | F5 | 264.256 | 1.75E+08 | 4.66E+08 | 1177.075 | 5.56E+08 | 4.64E+09 | 2.26E+09 | **51.84695** | 1.83E+09 | 6.33E+09 | 4.99E+09 | 3.38E+09 | 34607332 |
| Stddev | | 238.256 | 46027881 | 63386925 | 1700.292 | 1.22E+08 | 5.63E+08 | 1.22E+09 | **124.7717** | 4.71E+08 | 2.35E+08 | 6.35E+08 | 4.53E+08 | 3586676 |
| Rank | | 2 | 5 | 6 | 3 | 7 | 11 | 9 | **1** | 8 | 13 | 12 | 10 | 4 |
| Mean | F6 | 0.000842 | 89099.99 | 166363.5 | 79.69855 | 369891.9 | 504558.3 | 809805.5 | 1.391101 | 436549.8 | 1414570 | 1091100 | 943978.6 | 88008.73 |
| Stddev | | 0.004057 | 9995.932 | 12878.27 | 13.97079 | 39936.67 | 238019.1 | 226307.4 | 2.02195 | 30440.08 | 29925.42 | 33499.06 | 95597.17 | 4369.101 |
| Rank | | 1 | 5 | 6 | 3 | 7 | 9 | 10 | 2 | 8 | 13 | 12 | 11 | 4 |
| Mean | F7 | **0.001904** | 1142.403 | 60900.38 | 0.217079 | 4841.527 | 37967.51 | 25682.73 | 0.003659 | 13780.71 | 50303.71 | 36161.52 | 27028.82 | 244.6727 |
| Stddev | | **0.001606** | 279.4263 | 2537.809 | 0.291563 | 988.7598 | 5304.275 | 17311.93 | 0.0024 | 3775.191 | 2256.083 | 3973.04 | 4901.311 | 36.04973 |
| Rank | | 1 | 5 | 13 | 3 | 6 | 11 | 8 | 2 | 7 | 12 | 10 | 9 | 4 |
| Mean | F8 | -118634 | -19117.4 | -9330.34 | -132737 | -27768.5 | -12143.1 | **-6.38E+34** | -169777 | -21467.8 | -38402.84 | -25606.6 | -34052.9 | -33709.2 |
| Stddev | | 35419.56 | 8181.43 | 1802.173 | 20236.93 | 4844.248 | 1157.836 | **3.17E+35** | 34616.06 | 3467.874 | 3349.389 | 3595.01 | 6525.792 | 3454.41 |
| Rank | | 4 | 11 | 13 | 3 | 8 | 12 | **1** | 2 | 10 | 5 | 9 | 6 | 7 |
| Mean | F9 | **0** | 4289.875 | 7961.693 | 31.57649 | 5609.46 | 2145.167 | 7042.175 | 4.85E-13 | 5863.115 | 7808.152 | 7255.662 | 6660.146 | 4160.443 |
| Stddev | | **0** | 361.9354 | 198.1582 | 159.5589 | 152.9516 | 765.8859 | 808.8577 | 1.48E-12 | 246.6993 | 173.8322 | 193.3377 | 269.6606 | 94.92816 |
| Rank | | 1 | 6 | 13 | 3 | 7 | 4 | 10 | 2 | 8 | 12 | 11 | 9 | 5 |
| Mean | F10 | **6.66E-10** | 13.39598 | 17.7011 | 0.006042 | 18.13774 | 19.62215 | 20.09581 | 6.99E-08 | 19.53793 | 20.69109 | 20.58692 | 20.47253 | 13.17247 |
| Stddev | | **1.14E-09** | 0.467076 | 0.25372 | 0.00915 | 0.389211 | 2.498919 | 0.76774 | 1.87E-07 | 0.408294 | 0.053175 | 0.083081 | 0.178578 | 0.20436 |
| Rank | | 1 | 5 | 6 | 3 | 7 | 9 | 10 | 2 | 8 | 13 | 12 | 11 | 4 |
| Mean | F11 | **0** | 767.2542 | 1914.717 | 0.101973 | 3327.474 | 5133.207 | 7509.669 | 1.43E-11 | 3986.341 | 12641.4 | 9711.518 | 8490.546 | 793.9827 |
| Stddev | | **0** | 89.91628 | 133.862 | 0.228508 | 504.3143 | 1763.481 | 2712.571 | 7.31E-11 | 398.5241 | 265.7952 | 443.0763 | 761.8505 | 48.41929 |
| Rank | | 1 | 4 | 6 | 3 | 7 | 9 | 10 | 2 | 8 | 13 | 12 | 11 | 5 |
| Mean | F12 | **1.22E-05** | 1.98E+08 | 84664222 | 1988.757 | 7.65E+08 | 1.32E+10 | 7.88E+09 | 0.001831 | 5.33E+09 | 1.54E+10 | 1.29E+10 | 7.65E+09 | 1734358 |
| Stddev | | **3.05E-05** | 60844254 | 22058721 | 10658.64 | 2.97E+08 | 1.62E+09 | 6.34E+09 | 0.003726 | 2.51E+09 | 6.29E+08 | 2.13E+09 | 1.75E+09 | 853369.6 |
| Rank | | 1 | 6 | 5 | 3 | 7 | 12 | 10 | 2 | 8 | 13 | 11 | 9 | 4 |
| Mean | F13 | 1.007196 | 5.02E+08 | 4.6E+08 | 122.4901 | 2.01E+09 | 2.26E+10 | 9.57E+09 | **0.466298** | 8.87E+09 | 2.86E+10 | 2.42E+10 | 1.46E+10 | 33779145 |
| Stddev | | 1.76349 | 1.53E+08 | 1.4E+08 | 160.0758 | 6.04E+08 | 2.08E+09 | 7.12E+09 | **0.878845** | 4.18E+09 | 1.12E+09 | 2.97E+09 | 2.48E+09 | 7373217 |
| Rank | | 2 | 6 | 5 | 3 | 7 | 11 | 9 | **1** | 8 | 13 | 12 | 10 | 4 |
| average rank | | **1.384615** | 5.538462 | 7.538462 | 4.076923 | 6.923077 | 9.384615 | 9.230769 | 1.846154 | 8.692308 | 11.46154 | 10.92308 | 9.538462 | 4.461538 |
| final rank | | **1** | 5 | 7 | 3 | 6 | 10 | 9 | 2 | 8 | 13 | 12 | 11 | 4 |

## V. RESULTS

The experimentation for the proposed algorithm is conducted under three phases. The first phase uses 13 CEC 2014 benchmark functions of uni-modal and multi-modal functions under both small and large scale optimization to demonstrate the stability of the algorithm. The second phase uses MSCHHO optimized neural network with Quasi L1 regularization (MSCHHHO-NN-QL1) for classification over generic small dimensional medical datasets. The third phase uses MSCHHO-NN-QL1 for obtaining the biomarker for ASD using the large dimensional ABIDE dataset.

### A. *Testing over CEC-2014 benchmark functions:*

The proposed MSCHHO algorithm was tested over 13 standard CEC 2014 benchmark functions on 500 dimensions. The algorithm was run for 1000 iterations with 30 agents, and the fitness comparison is given in the table1. The proposed algorithm has obtained 10 1$^{st}$ ranks out of the 13 benchmark functions. The proposed algorithm has out beaten the GWO, PSO, WOA, CS, SCA, JAYA, MFO, FFA, DE, and SSA algorithms on 100% of the benchmark functions and BAT algorithm over 92% of the functions and HHO algorithm over 84% of benchmark functions. The proposed MSCHHO algorithm has achieved zero fitness over two benchmark functions F9 and F11.

The MSCHHO has achieved near absolute zero in 8 of the benchmark functions. And even with the three benchmark functions, it has not achieved the first position, but it has performed well to obtain the optimal position. On average, from table 1, it is clear that the proposed algorithm has performed well and is being robust to all conditions.

### B. *Classification of medical datasets*

This subsection uses the proposed MSCHHO-NN algorithm for the classification of breast cancer using a generic Wisconsin dataset and autism spectrum disorder using ABIDE dataset. Totally 30 agents were used to train the NN in both experiments

#### *a) Breast cancer classification*

For the classification of breast cancer over the Wisconsin dataset, the neural network architecture of one input layer, one hidden layer of 10 neurones, and an output layer were being used. A total of 100 iterations for training the neural network over each experiment was carried out. A weight of 0.0001 was given to the quasi L1 regularization. The data set was split into 80% for training and 20% for testing.

The proposed algorithm was able to obtain the highest accuracy of 100% and an average accuracy of 99.3% over 30 iterations of the experiment. Comparing to other existing algorithms, the proposed algorithm achieved paramount accuracy. The next highest accuracy was obtained by the MSCHHO algorithm with ordinary L1 regularization is of 98.9%.

This shows the usage of Quasi L1 regularization that has given a considerable boost for classification by choosing only the necessary features from the input layer. The naïve HHO algorithm has achieved an accuracy of 98.6%, which is the third-highest, followed by the backpropagation algorithm. This shows the efficiency of the proposed algorithm, which has out beaten the backpropagation training algorithm for neural networks and found a more optimal solution for classifying. The salp Swarm algorithm, which uses the chaining behavior, has achieved the lowest accuracy of 66%. From table 2, it is evident that the standard deviation of the proposed algorithm has obtained the least standard deviation and thereby proven to be the robust algorithm.

#### *b) Autism spectrum disorder classification*

This subsection uses the proposed algorithm to identify the biomarkers for ASD disease. For identifying the biomarker NN architecture of one input layer, one hidden layer of 200 neurons and a final output layer were proposed. Similar to the breast cancer dataset, the split was 80% for training and 20% for testing. After pre-processing and the feature importance selection, the final feature is subjected to classification using the MSCHHO-NN algorithm. The proposed MSCHHO-NN algorithm has obtained an accuracy of 76.3% over the classification of ASD. The other bio-inspired algorithms were also utilized for training the NN algorithm with the same configuration, and it is observed from the table that all the other algorithms obtained lower frequencies. The MSCHHO-NN with ordinary L1 regularization and ANN with L1 regularization were also trained on the same feature set and obtained the accuracies that are less than the proposed algorithm. This proves the robustness of the proposed algorithm on the task of classification.

## VI. DISCUSSION

The MSCHHO algorithm being tested over the benchmark function and the classification tasks over the medical datasets have proved to be the best.

**Table 1 Accuracy comparison for breast cancer diagnosis**

| | QL1 – Regularization | | | | | | | | | | | L1 - Regularization | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Algo** | **BAT** | **SSA** | **SCA** | **HHO** | **PSO** | **MSC - HHO** | **WOA** | **MVO** | **FFA** | **GWO** | **MFO** | **MSCHHO** | **backprop** |
| **Acc-mean** | 0.666667 | 0.657143 | 0.782381 | 0.986667 | 0.982857 | **0.99381** | 0.92952 | 0.671905 | 0.666667 | 0.69381 | 0.70619 | 0.98908 | 0.98574 |
| **Acc-std** | 0.051287 | 0 | 0.134957 | 0.011025 | 0.013997 | **0.0072** | 0.05283 | 0.060213 | 0.03639 | 0.095382 | 0.102929 | 0.00894 | 0 |
| **Rank** | 11 | 13 | 7 | 3 | 5 | **1** | 6 | 10 | 11 | 9 | 8 | 2 | 4 |

**Table 3 Accuracy comparison for ASD using ABIDE dataset**

| | QL1 | | | | | | | | | | | L1 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Algo** | **MSCHHO** | **GWO** | **MVO** | **SSA** | **WOA** | **PSO** | **MFO** | **SCA** | **FFA** | **BAT** | **HHO** | **MSCHHO** | **backprop** |
| **Acc-mean** | **0.763333** | 0.513333 | 0.5 | 0.511111 | 0.568889 | 0.555556 | 0.513333 | 0.504444 | 0.504444 | 0.5 | 0.584444 | 0.632 | 0.628571 |
| **Acc-std** | **0.077905** | 0.075326 | 0.033333 | 0.031427 | 0.070413 | 0.077619 | 0.030551 | 0.053564 | 0.033036 | 0.033333 | 0.112458 | 0.090521 | 0.075506 |
| **Rank** | **1** | 9 | 12 | 11 | 5 | 6 | 9 | 7 | 7 | 12 | 4 | 2 | 3 |

Apart from the bio insipid algorithms, there has been recent work in the literature that has diagnosed ASD using the fMRI. The authors [20] have utilized an autoencoder for the feature space reduction and later utilized the MLP algorithm for classification to acquire the highest accuracy of 74.5%. In this work, instead of using features from single atlases, the authors have aggregated them from multiple atlases. The work [21] presented a study of the prevalence of autism based on gender heterogeneity, severity, and overlapping. The authors utilized a random forest classifier in this work, and the highest accuracy of 73.55 was obtained. The authors [22] in their work encoded that the correlation and phenotypic features over the nodes and edges of the graph and used Graph CNN for classification. An accuracy of 70.4% was obtained on the utilization of Graph CNN. In the proposed model, the pipeline of feature selection is a hybrid feature selection comprising of random isolation forest as a filter method and Quasi-L1 regularization, which is an embedded method along with the effective training algorithm enabled to achieve good accuracy of 76.3%.

### *a) Limitations*

The proposed algorithm has utilized an rs-fMRI-based dataset hasn't utilized any other image modality. The proposed algorithm has also not utilized any other atlas than the CC200. In future work, a much stronger and robust classifier can be built incorporating all the heterogeneous data.

## VII. CONCLUSION

The work investigated the existing and recently proposed HHO algorithm and thereby proposed a Multi Swarm variant of the algorithm that parallelizes and increases the efficiency of the HHO algorithm. The proposed Multi Social Communicative HHO uses the Multi Swarm initialization and then socializes its behavior for joining together to perform a unified search. At the same time, the weak population in the Swarm is also being enhanced by the strong population by enabling communication among the swarm members. The proposed MSCHHO algorithm has been used to train the NN architecture and tested over two different datasets to prove its robustness. The proposed algorithm has achieved an accuracy of 99.3% over breast cancer classification and 76.3% over the ASD classification. The accuracy results were compared with the existing bio-inspired and recently proposed algorithms.

## REFERENCES


[1] M. C. Lai, M. V. Lombardo, and S. Baron-Cohen, Autism, in The Lancet, 383(2014) 9920 896–910.

[2] P. A. Filipek et al., The Screening and Diagnosis of Autistic Spectrum Disorders 1 (1999).
[3] C. Qu, W. He, X. Peng, and X. Peng., Harris Hawks optimization with information exchange, Appl. Math. Model., 84(2020) 52–75.

[4] B. Niu, H. Huang, L. Tan, and Q. Duan., Symbiosis-Based Alternative Learning Multi-Swarm Particle Swarm Optimization., IEEE/ACM Trans. Comput. Biol. Bioinforma., 14(1) (2017) 4–14.

[5] H. Xiong, B. Qiu, and J. Liu., An improved multi-swarm particle swarm optimizer for optimizing the electric field distribution of multichannel transcranial magnetic stimulation., Artif. Intell. Med. 104(2019) (2020) 101790.

[6] S. Rekha, D. Debahuti, and M. Minakhi, "A hybridized ELM using self-adaptive multi-population-based Jaya algorithm for currency exchange prediction : an empirical assessment., Neural Comput. Appl., 8(2018).

[7] S. S. Tirumala., Evolving deep neural networks using coevolutionary algorithms with multi-population strategy., Neural Comput. Appl., 32(16) (2020) 13051–13064.

[8] Y. Ma., A multi-population differential evolution with best-random mutation strategy for large-scale global optimization,( 2019).

[9] A. S. Heinsfeld, A. R. Franco, R. C. Craddock, A. Buchweitz, and F. Meneguzzi., Identification of autism spectrum disorder using deep learning and the ABIDE dataset., NeuroImage Clin., 17(2018) 16–23.

[10] A. El-Gazzar, M. Quaak, L. Cerliani, P. Bloem, G. van Wingen, and R. Mani Thomas., A Hybrid 3DCNN and 3DC-LSTM Based Model for 4D Spatio-Temporal fMRI Data: An ABIDE Autism Classification Study, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 11796 LNCS, (2019) 95–102.

[11] B. Tran, B. Xue, and M. Zhang., Variable-Length Particle Swarm Optimisation for Feature Selection on High-Dimensional Classification, IEEE Trans.. Comput., 1(2018).

[12] H. Faris, I. Aljarah, M. A. Al-Betar, and S. Mirjalili, "Grey wolf optimizer: a review of recent variants and applications., Neural Computing and Applications, vol.. Springer London, 30(2) (2018) 413–435.

[13] S. Mirjalili and A. Lewis., The Whale Optimization Algorithm., Adv. Eng. Softw., 95(2016) 51–67.

[14] N. B. Arunekumar, A. Kumar, and K. S. Joseph., Hybrid bat-inspired algorithm for multiprocessor real-time scheduling preparation, 2016 Int. Conf. Commun. Signal Process., (2016) 2194–2198.

[15] C. Paper., An improved cuckoo search algorithm for parallel machine scheduling Metadata of the chapter that will be visualized in SpringerLink, (2015).

[16] J. Wu, Y. G. Wang, K. Burrage, Y. C. Tian, B. Lawson, and Z. Ding., An improved firefly algorithm for global continuous optimization problems, Expert Syst. Appl., 149( 2020).

[17] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems, Adv. Eng. Softw., 114(2017) 163–191.

[18] A. Bedri Ozer and Ahmet., CIDE: Chaotically Initialized Differential Evolution, Expert Syst. Appl., 37(6) (2010) 4632–4641.

[19] Y. Li, Y. Zhao, and J. Liu., Dynamic sine cosine algorithm for large-scale global optimization problems, Expert Syst. Appl., (2021) 114950.

[20] Y. Wang, J. Wang, F. X. Wu, R. Hayrat, and J. Liu, AIMAFE: Autism spectrum disorder identification with multi-atlas deep

feature representation and ensemble learning, J. Neurosci. Methods, 343 (2020).

[21] M. A. Reiter, A. Jahedi, A. R. J. Fredo, I. Fishman, B. Bailey, and R. A. Müller., Performance of machine learning classification models of autism using resting-state fMRI is contingent on sample heterogeneity, Neural Comput. Appl., 33(8) (2021) 3299–3310.

[22] S. Parisot et al., Disease prediction using graph convolutional networks: Application to Autism Spectrum Disorder and Alzheimer's disease., Med. Image Anal., 48(2018) 117–130.