International Journal of Psychosocial Rehabilitation, Vol. 24, Issue 06, 2020
ISSN: 1475-7192

FEATURE SELECTION USING BIO INSPIRED ALGORITHMS

[1]Neeyati Anand, [2]Shallu Sehgal, [3]Manisha Agarwal,[4]Riya Sehgal,[5]Sanchit Anand, [6]Arun Bashambu

ABSTRACT-- *Bio-inspired computing optimization algorithms are recently developed algorithms which are influenced by the biological progression of nature. These algorithms are proving to be better than the traditional machine learning algorithms as they can determine optimal solution of complex problems in the field of science. This paper presents the 10 recent bio-inspired algorithms as well as their diverse applications, especially in the medical field namely- Artificial Bee Colony (ABC) Algorithm, Fish Swarm Algorithm (FSA), Cat Swarm Optimization (CSO), Whale Optimization Algorithm (WOA), Artificial Algae Algorithm (AAA), Cuttlefish Algorithm (CFA), Bat Algorithm(BA), Grasshopper Optimization Algorithm(GOA), Ant Lion Optimization Algorithm(ALO) and Crow Search Algorithm (CSA). We are trying to analyze the ways through which they mimic evolutionary operators. Also, we study and analyse the results with the help of feature selection using bio-inspired algorithms. It is an essential process that is relevant for predictive analysis. It is considered to be the most important step before using machine learning algorithms. Crow Search algorithm is used for feature selection and gives an accuracy of 98% by using Support vector machine(SVM).*

Keywords---*Crow Search Algorithm; Feature Selection; Fish Swarm Algorithm; Optimization; Swarm Intelligence.*

I. INTRODUCTION

Bio-inspired computing is the umbrella of diverse studies of computer science, mathematics, and biology and has been studied a lot in the last few years. Bio-inspired computing optimization algorithms are constructed on the principles and are motivated by the biological advancement of nature. New challenging techniques are developed using this method [35]. Traditional algorithms do not produce better results for complex optimisation problems. Now a day's enormous bio-inspired heuristic algorithm are emerging to solve optimisation problems (Baghel, Malti, Shikha Agrawal, and Sanjay Silakari, 2012) [39]. These problems are usually nonlinear and produce multiple problems such as taking much time in finding optimal solutions. To deal with such problems of the optimization algorithms, recent trends apply bio-inspired optimization algorithms which gives a promising method for solving complex optimization problems. Bio-inspired algorithms have applications in medical, neural networks and many other fields [43][44]. These algorithms will become more effective in solving various other problems in

[1] ***B Tech(CSE) Student Maharaja Agrasen Institute of Technology, Rohini, Delhi, India***
[2] ***Research Scholar, Banasthali Vidyapeeth, Banasthali, Jaipur, India***
[3]***Associate Professor, Banasthali Vidyapeeth, Banasthali, Jaipur, India***
[4]***B Tech(CSE) Student Maharaja Agrasen Institute of Technology, Rohini, Delhi, India***
[5] ***B Tech(CSE) StudentMaharaja Agrasen Institute of Technology, Rohini, Delhi, India***
[6] ***Research Scholar, Banasthali Vidyapeeth, Banasthali, Jaipur, India***

the engineering field as well**.** Global optimization algorithms diverge into two parts- deterministic and meta-heuristic [3]. These are explained below:

1. Deterministic algorithms are known for their ability to solve unimodal problems by a gradient technique while meta-heuristic models are adaptive and learn as they run.

2. The meta-heuristic method can find an optimum solution very fast as compared to a deterministic one. Intensification (exploitation) and diversification (exploration) are the constituents of metaheuristic. Diversification involves the task of traversing the search space for all possible solutions while intensification is to look into a particular solution. A good balance between the two assures the selection of best solutions.

3. Intensification, also called exploitation, is the phase that finds the best solutions. The diversification, also called exploration is the phase that keeps track of whether the algorithm explores the search space more efficiently [7]. The balance between these two factors decides the efficiency of an algorithm. If there is too much exploration but too little exploitation, it becomes difficult for the system to converge. Hence, balancing these two is the most crucial task for optimization.

4. The main source of inspiration is from nature itself. The two diverse categories of NIA have made a positive effect and have inspired many to use nature-inspired approach into their field. NIA is divided into- Bio-Inspired Algorithms and Swarm Intelligence Algorithms. In fact swarm intelligence is associated with a broader class of bioinspired algorithm that creates the majority of nature-inspired algorithms.

5. Swarm intelligence is a subset of bio-inspired algorithms and bio-inspired is further, a subset of nature-inspired algorithms [38]. Some of the applications of bio-inspired algorithms include feature selection, feature extraction, segmentation, scheduling, control systems, routing, etc [39].

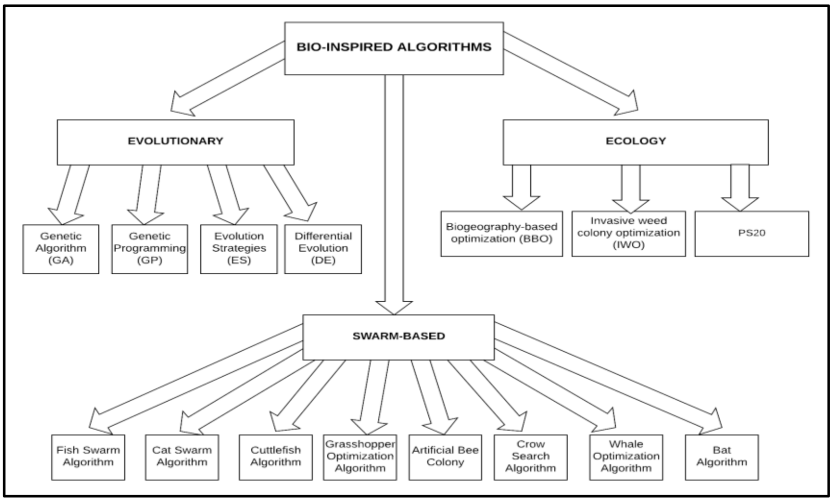

Figure 1: Classification of Bio-Inspired Algorithms

II. SWARM-INTELLIGENCE-BASED ALGORITHMS

Swarm-Intelligence-Based Algorithms have inherited the properties of many animals such as birds, fish, and insects. Collective intelligence is another term for swarm intelligence. Many types of research have been made on the behaviour of insects by the researchers. It is done because of their ability to solve complex processes. The insects can create wonders in a swarm. Swarm intelligence is a part of artificial intelligence which is bothered with the intelligent behavior of biological swarms with the help of interaction among individuals in such environments.

Such behaviour is then used to solve real-world problems. Ants and insect colonies are capable of providing a great environment to develop bio-inspired algorithms [35].

III. BIO-INSPIRED SWARM OPTIMIZATION ALGORITHMS

Some of these algorithms and how they use evolutionary operators are explained below:

A. Artificial bee algorithm colony (ABC)

Mainly three types of bees that are employed bees, onlooker bees, and scout bees are there in the ABC algorithm. When an employed bee finds a food source, it stores its location in its memory. Once all the employed bees are done searching for the food sources, they return to the hive. They start dancing in the dance area which is a way for communication among the bees and the dance is known as the waggle dance. The onlooker bees look at the dances of the employed bees to gain information about the food sources. Based on the dance, they decide which food source has the highest amount of nectar in it so that they can exploit it. When the employed bees are done exchanging information, they start looking for a new source. They replace its position with the previous one only if the amount of nectar is more than the previous one.

Some of the advantages of the artificial bee algorithm colony are its simplicity, flexibility, easy implementation, robustness, and popularity. Its ability to explore local solutions and to handle objective cost makes it an ideal algorithm. On the other hand, it has some limitations too such as it requires a higher number of objective function evaluations, slow when in sequential processing, and lack of use of secondary information.

In the artificial bee colony algorithm, a random initial population of SN solutions is produced where SN is half of the swarm size. Let $x_{ij} = \{x_{i,1}, x_{i,2}, \ldots, x_{i,n}\}$ represent the i^{th} solution i.e. food source in the swarm and n is the dimension size. Each food source is produced within the limited range of jth index by [31]:

$$x_{ij} = x_j^{min} + \lambda(x_j^{max} - x_j^{min}), \qquad (1)$$

Where i = 1,2,..., SN and j=1,2,…,N, , x_j^{max} and x_j^{min} are the upper and lower bounds for the index j, respectively and λ has the value between 0 and 1.The candidate solution is produced by the employed bees by the help of a local search around a neighbouring food source. Its equation is given below [30]:

$$V_{ik} = X_{ik} + \varphi_{ik} \times (X_{ik} - X_{jk}) \qquad (2)$$

Where x_{ij} is a randomly selected candidate solution (i≠j), k is a random dimension index selected from the set {1,2,..,n), and $\phi_k$is a random number within [-1,1]. Whenever a new candidate solution V_i is generated, a greedy selection is used.If the fitness value of $V_i$is better than that of its parent X_i, then update V_i with X_i otherwise keep unchanged. Once the search process of the employed bees is complete, they start sharing the information of their food sources with the onlooker bees. The onlooker bees then take a look at the information and select that food source whose probability is related to its nectar amount. the probabilistic selection is very much similar to a roulette wheel selection mechanism which is given below: [32]:

$$P_i = \frac{fit_i}{\sum fit_j} \quad (3)$$

Where fit_i is the fitness value of the ith solution in the swarm, the better the solution i , the higher probability of the ith food source is chosen as the solution i gets better. If a position is not progressing over a predefined number (called limit) of cycles, then the food source is abandoned. Assume that the abandoned source is X_i, and then the scout bee discovers a new food source to be replaced with ith as equation below [32]:

$$X_{ik} = lb_j + \text{rand}(0,1) \times (ub_j - lb_j) \quad (4)$$

Where rand(0,1) is a random number within [0,1] based on a normal distribution and lb, ub are lower and upper boundaries of the i^{th} dimension, respectively.

➢ Advantages:

1. It is a simple, flexible and robust algorithm.
2. Widely used.
3. Can easily handle objective cost.
4. Easy to implement.
5. Can explore local solutions properly.
6. Applications in various domains.

➢ Limitations:

1. No secondary information is required.
2. Need new fitness tests on new algorithm parameters.
3. Need higher number of objective function evaluations.
4. Slow Algorithm.

➢ Applications:

1. To overcome local optima problem.
2. Multi-objective layout optimization.
3. Network optimization.
4. Healthcare.

B. Fish swarm optimization algorithm (FSOA)

The fish swarm optimization algorithm imitates the behavior of fish and is used for making predictions. A fish can look for its food in many ways. It can do so either by individual search or by following other fish. An area with a large number of fishes generally represents a nutritional source for the fish. When a fish explores the search space for its food, it is known as the exploration phase. The fish always try to maintain their colonies and communicate in a group which results in intelligent social behavior.

Some of the characteristics which are kept in mind while developing the algorithm are the objective function which is inversely proportional to the amount of food in a region, the sample space and the fish which themselves represent as candidate solutions for the optimization algorithm.

IV. INDIVIDUAL MOVEMENT OPERATOR

This operator contributes to 't' individual and collective movements of fish in the swarm. Each fish changes its new position by using the following equation:

$$x_i^{t+1} = x_i^t + \text{rand} \times s_{ind}$$

(5)

where x_i is the final position of fish i at current generation, rand is a random generator and s_{ind} is a weighted parameter.

V. FOOD OPERATOR

The success of food search is determined by the weight of each fish. The higher the weight of a fish, the more likely this fish will be in a potentially interesting region in the search space. The amount of food that a fish eats depends on the improvement in its objective function in the current generation. The weight is modified as :

$$W_i^{t+1} = W_i^t + \frac{\Delta f}{max(\Delta f)}$$

(6)

where W_i^t is the fish weight *i* at generation *t* and Δ f is the difference of the objective function between the current position and the new position of fish 'i'. The fish is at the same position when $\Delta f_i=0$.

VI. INSTINCTIVE COLLECTIVE MOVEMENT OPERATOR

When $\Delta fi \neq 0$, this operator becomes important for the individual movement of fish. The fishes whose individual execution of the movement enhances their fitness will affect the direction of motion of the school that will result in an instinctive collective movement. The resulting direction ($\vec{I}^t$), calculated using the involvement of the directions taken by the fish, and the new position of the ith fish is given by [34]:

$$\vec{I}^t = \frac{\sum_{i=1}^{N} \Delta \vec{x}_i \Delta f_i}{\sum_{i=1}^{N} \Delta f_i}$$

(7)

$$\vec{x}_i^{t+1} = \vec{x}_i^t + \vec{I}^t$$

(8) The swarm is directed by the instinctive collective movement.

VII. NON-INSTINCTIVE COLLECTIVE MOVEMENT OPERATOR

The search success for food is determined using the fish weight. Due to this reason, it is assumed that when the swarm weight is increasing, the search process is performing successfully. So, the "radius"of the swarm must decrease so that other regions can be examined. Otherwise, if the swarm weight remains constant, the radius should increase to allow the exploration of new regions. For the swarm contraction, the centroid concept is used. This is accomplished by means of an average position of all fish weighted with the respective fish weights [34] :

$$\vec{B}^t = \frac{\sum_{i=1}^{N} \vec{x}_i W_i^t}{\sum_{i=1}^{N} W_i^t}$$

(9)

All the fishes should alter their location if the swarm weight remains fixed in the current iteration which is represented as:

$$\vec{x}^{t+1} = \vec{x}^t - s_{vol} \times rand \times \frac{\vec{x}^t - \vec{B}^t}{d\left(\vec{x}^t, \vec{B}^t\right)}$$

(10)

where d denotes a function which computes the Euclidean distance between the present position of fish and the centroid. s_{vol} represents the step size required for controlling fish displacements.

- Advantages:

1. Robust method of optimization.
2. have a better shot at getting good answers with more particles.
3. Global Search Ability.
4. Tolerance of parameter setting.

- Disadvantages:

1. Higher time complexity.
2. Low convergence speed.
3. Experience of group members is not used in the next iteration.

- Applications:

1. Software testing.
2. Synchronous optimization.
3. Packet routing.
4. Fault identification.

C. Cat swarm optimization (CSO)

CSO algorithm is an optimization algorithm that can mimic the cat's behavior. CSO has been applied to many problems in order to find an optimal solution [8]. It has two modes that are seeking and tracing modes. The resting phase involves the seeking mode and the tracing mode and are used to discover the optimal solution of the given problem. The position of a cat comprises M dimensions and each dimension has a velocity associated with it. The flag tells about the mode of cat whether the cat is in seeking more or a tracing mode.

- ***Population Initialization***

Before starting the process, deciding the number of individuals, i.e, the cats that will take part in the optimization process is important. The position of the cat represents the candidate solution and the fitness value of each cat represents the accommodation of the cat to the fitness function. The seeking/tracing flag is required to recognize whether the cat is in seeking mode or tracing mode [15]. The initial population of cats within the solution search space is initialized as follows:

$$x_{id}^{t} = x_d^{min} + \mathrm{R}(x_d^{max} + - x_d^{min}); \ (\text{d}= 1,2,\ldots,\text{D}, \text{i}=1,2,\ldots,\text{T}) \quad (11)$$

And the velocity for each dimension is mathematically given as:

$$v_{id}^{t} = v_d^{min} + \text{c} \times \mathrm{R}(x_d^{min} - x_d^{min}); \ (\text{d}= 1,2,\ldots,\text{D},\text{i}=1,2,\ldots,\text{T}) \quad (12)$$

where x_{id}^{t} and v_{id}^{t}, represents the position and velocity of the ith cat in dth dimension, respectively. C is a constant and R is a uniform random number between the range [0, 1]. The population may violate inequality constraints which are corrected by applying the random perturbation method.

- ***Fitness Evaluation***

The aim of the optimization process is to minimize the objective function. There is a possibility of the elements of parent/offspring to violate the constraint. Therefore, a penalty term is introduced and the objective function is penalized and changed to a generalized form which is mathematically expressed as follows:

$$A_i(X_i) = e_i(X_i) + \mathrm{R}\,(P_{term}) \ (\text{i}=1,2,\ldots,\text{T}) \quad (13)$$

- ***Seeking Mode***

The seeking/tracing flag is set by a mixture ratio which is responsible for the number of cats that would randomly be placed into the seeking mode and the tracing mode. The seeking mode corresponds to the global search procedure and this mode creates copies of the current solution. When the copies are done exploiting the current solution, the new position is chosen in which the cat has to move. The factors that are crucial for the seeking mode are - Memory Seeking Pool (MSP), Seeking Range of Dimension (SRD), Counts of Dimension to Change (CDC) and Self-position consideration (SPC). These are explained below:

1. MSP gives the size of seeking memory for each cat.
2. SRD help in selecting a dimension. If a dimension is adopted to mutate then its value must be in the domain specified by the SRD.
3. The number of dimensions that can be varied is given by the CDC.
4. The candidate can move to a point where the cat is standing only when the SPS allows it to do so.

't' copies of the present position of an ith cat are generated by the seeking mode and t=MSP. If SPC=1 then let t=MSP-1. For each copy, SRD percents the present value is either subtracted or added. The fitness of all copies is evaluated when the seeking mode ends and from t copies, that candidate is chosen who has the highest fitness value.

- Tracing mode

The tracing mode includes the mathematical exemplar of the cat chasing its prey.

➢ Advantages:

1. Has the ability to find the global optimal solution and consistent solution.
2. Is known for its efficiency and precision for solving the problem.

➢ Limitations:

1. Can get trapped in the local area.
2. More efficient for a small number of particles.

D. Whale optimization algorithm (WOA)

The Whale optimization algorithm (WOA) is a meta-heuristic algorithm. It focuses on attacking the population. This algorithm imitates bubble-net attacking behavior of the whales when they search for their prey.

- ***Encircling prey***

Humpback whales are known for acknowledging the location of their prey and encircle them accordingly. The exact location of the prey is not known exactly, so the WOA algorithm assume the current best prospect is the target prey or the closest one. After the best search agent gets explicated, the other search agents will now try to update their positions towards the best one. This functioning is constituted by the following equations:

$$D= |C.X^{*}(t) - X(t)| \quad (14)$$

$$X(t+1)= X^{*}(t)-A.D \quad (15)$$

Where t represents the current iteration, A and C are coefficient vectors, X^{*} denotes the location vector of the best solution and X signifies the position vector of the solution. $| |$ denotes the absolute value [8]. The two vectors A and C are deliberated as follows:

$$A=2a.r_1 - a \quad (16)$$

$$C=2.r_2 \quad (17)$$

Where components of r is decreasing linearly from 2 to 0 over the course of iterations and r is random vector [0;1].

- ***Search for prey:***

The whales usually search randomly amongst the population. Therefore, we use A with the random values greater than 1 or less than -1 to constrain the search agent to move away from the reference whale. In variance to the exploitation phase, the position of the search agent gets updated in the exploration phase according to a randomly chosen search agent instead of the best search agent. The mathematical model is as follows:

$$D = |C.X_{rand} - X| \quad (18)$$

$$X(t+1) = X_{rand} - A.D \quad (19)$$

Where X_{rand} is a random position vector chosen from the current population.

➢ Advantages:

1. This algorithm solves large scale optimization problems.

➢ Applications

1. Diagnosis of Breast cancer.
2. To recognize Arabic Handwritten Characters.
3. Liver segmentation in MRI images.

E. Artificial algae algorithm (AAA)

AAA is a bio-inspired algorithm, and it impersonates the lifestyles of microalgae. This algorithm is based on microalgae behavior such as the algal tendency, reproduction, and their adaptation to the surrounding environment to change the species. Algae has three important processes called evolutionary process, helical movement, and adaptation. Algal colonies are the main components of the population. The algal cells grow bigger in size when they receive enough light and accordingly the algal colonies also enlarge. In the budding phase, they may not grow that much due to insufficient light. Whereas in the helical movement, all the algal colonies are able to move in the direction of the best algal colony.

As Algae stays close to the surface of water for them to receive ample amounts of light, they are known for their good swimming skills. In AAA the global optimum of the objective function was defined to the point on which algae can receive optimum light for photosynthesis [10]. Artificial algae corresponds to each solution in the problem space by representing the features of algae. Alike to the real algae, artificial algae can also move towards the origin of light to photosynthesize with the helical swimming.

The algorithm has three major processes which are- Evolutionary Process, Adaptation and Helical Movement.

1. **Evolutionary Process -** The evolutionary process elucidates the growth features of the algae which can be given by Monod equation

$$G_j^{t+1} = \left(\frac{f(x)}{k + f_j(x)} * G_j^t\right), (j = 1, 2, \ldots N) \quad (20)$$

where G represents the size of j^{th} algal colony in time t, N denotes the number of algal colonies in the system. Algal colonies provide cost-efficient solutions as they grow fast when the amount of nutrients that they obtain is high. Replication of the algal cell of the biggest algal colony is done with the dying algal cell of the smallest algal colony.

$$biggest^t = \max(G_j^t), (j = 1, 2, \ldots N) \quad (21)$$

$$smallest^t = \min(G_j^t), (j = 1, 2, \ldots N) \quad (22)$$

$$smallest_m^t = biggest_m^t, (m = 1, 2, \ldots D) \quad (23)$$

D is the problem dimension.

2. **Adaptation Process -** When an inadequately grown algal colony tries to resemble itself to the biggest algal colony in the environment is commonly referred to as Adaptation. And this results in the change in starvation level of the algorithm. The value of initial starvation for each colony is zero. It increases with time t, when algal cells receive sufficient amounts of light [11].

$$starving^t = maxAj^t, (j = 1, 2, \ldots N) \quad (24)$$

$$starving^{t+1} = starving^t + (biggest^t - starving^t) * \text{rand} \quad (25)$$

Where At j indicates the starvation value of j^{th} algal colony for time t, and $starving^t$ denotes the algal colony having the greatest starvation value in time t. Adaptation parameter (Ap) determines whether thadaptation process would be relevant in time t or not. Ap is constant within the interval [0,1].

3. **Helical** Movement- The colonies of the algae can remain closer to the water surface for their survival only if there are sufficient light and algae cells. The only thing which hinders the process is gravity which is represented by 0.

$$T(x_j) = 2\pi \left(\frac{3G_i}{4\pi}\right)^{\frac{3}{2}} \quad (26)$$

This algorithm has been used for continuous optimization problems and numerical optimization.

F. Cuttlefish Algorithm(CFA)

This algorithm imitates the color amelioration behavior of cuttlefish to unravel the optimisation problems in the biomedical field. The colors on the skin of a cuttlefish changes rapidly, Iridophores and leucophores present in cuttlefish can reflect light across many wavelengths. Cuttlefish has the capability to combine various cells that allows them to exhibit a large assemblage of hues.

❖ Chromatophores: are a group of specialized cells which contain an elastic saccule that holds a colored pigment, as well as 15-25 radial muscles attached to this saccule. These muscles are supervised by neurons present in the brain and when they become activated, they cause the muscles to contract which in result displays the pigment. And the opposite phenomenon happens when the muscles relax.

❖ Iridophores: are found in the next layer under the chromatophores. Their main function is to reflect light falling in the visible spectrum. Iridophores have the properties of both plane mirrors as well as scattering surfaces.

❖ Leucophores: These are those cells on cuttlefish which are accountable for the white spots. They are flattened, branched cells that are present for scattering and reflecting the incoming light. This is the pattern of the color generation of leucophores that they will reflect the predominant wavelength of light in the surroundings, which signify that in white light they will be white, while in blue light they will be blue [39].

The Cuttlefish algorithm for the most part have two processes- Reflection and visibility. Reflection process is extended to simulate the light reflection phenomenon of cuttlefish, while the visibility process is advanced to simulate the visibility of harmonizing pattern used by the cuttlefish. These two processes are contemplated to explore for the optimal solution.The formulation of finding the new solution (newP) with the use of reflection and visibility is as follows:-

newP= Reflection + Visibility

(27)

➢ Advantages:

1. Cuttlefish Algorithm is very stable as it gives compatible results.

This algorithm has a wide range of applications in diverse fields-

1. computer networks.
2. Security.
3. Robotics.
4. biomedical engineering.
5. production engineering.

G. Bat Algorithm(BA)

Bat algorithm is a recently introduced metaheuristic optimization algorithm. It locates the objects by the reverberant sound by microbats that use fluctuating pulse rates of emission and loudness. This algorithm stabilises two parameters - the exploration which means long-range jumps which surround the global search space to avoid them getting stuck around one local maximum and exploitation which is persistent search about and around known good solutions to find local maxima by managing the loudness and pulse emission rates of imitating bats in multiple directions**.**

A. BEHAVIOUR OF MICROBATS

To find their prey they emit sonar signals, and these generated signals rebound back when they strike an object. Bats explicate these signals to determine the size of the object which they had smacked and calculate whether it is moving towards or away. Bats fly with a velocity $V_i$at position x_i with a constant frequencyf_{min} , fluctuating wavelength λ and loudness A_0 when they are hunting for their prey. The wavelength of their emitted pulses is altered and also, the rate of pulse emission rϵ[0,1] is selected according to the nearness of their target.

1. BAT MOTION

Fostering new solutions is executed by moving virtual bats according to the following equations:

$$f_l = f_{min} + (f_{max} - f_{min})\beta \quad (28)$$

$$v_i^t = v_i^{t-1} + (x_i^t - x^*)f_i \quad (29)$$

$$x_i^t = x_i^{t-1} + v_i^t \quad (30)$$

Where $\beta\epsilon[0,1]$ is a uniform distribution random vector

f_{min}=0 and f_{max}=100

x^* gives the current global best location (solution) which can be represented after comparing all the solutions among all the bats. Loudness starts to decrease as the rate of pulse of emission starts increasing. This occurs when the prey has been found by the bat.

2. RANDOM WALK

The local search is performed using a random walk that revises the current best solution according to equation:

$$x_{new} = x_{old} + \varepsilon A^t;\ \varepsilon \epsilon [-1,1] \quad (31)$$

where ε is the scaling factor, and A_i^t is the loudness. Launching the local search with proximity which depends on pulse rate r_i. Both attributes are imitating the natural bats, whereas the rate of the pulse emission increases and the loudness decreases when a bat finds its prey. Mathematically, these attributes can be represented as:

$$A_i^{t+1} = \alpha A_i^{(t)} \quad (32)$$

$$r_i^{(t+1)} = r_i^{(0)} * [1 - \exp(-\gamma \varepsilon)] \quad (33)$$

where α and γ are constants. Parameter A observes the convergence rate of the algorithm.

➢ Advantages:

1. This algorithm is simple which solves non- linear problems as well.
2. It triggers best optimal solutions.
3. Can solve complicated problems with much ease.
4. Quick Implementation.
5. The loudness and pulse emission rates provide a mechanism of automatic control over an area.

➢ Applications:

1. Content-Based image retrieval.
2. Healthcare.

H. ***Grasshopper Optimization Algorithm (GOA)***

Grasshoppers are insects which are known for damaging agriculture. They are also responsible for bringing down the crop production. They can fly as well as jump. The life cycle of the grasshopper begins from the egg state. As the egg hatches, it produces offspring which is more commonly known as the Nymph. Nymphs are slow as they can only walk on land. Their wings are not fully developed at this stage. As they cannot fly they eat all the vegetation that is around them. With the passing of time, they develop into adults. At this stage, their wings are fully developed and they can scout for food in a larger area. Earlier they had a small search space. They can also move much faster now. They now have the ability to form swarms to look out for food. Usually, the grasshoppers are seen individually in nature but can form the largest swarm of all the animals. A swarm of grasshoppers is known as locusts and it is very difficult to protect the crops from it. The farmers get most affected by it as it reduces the crop yield. Long-range and abrupt movement is the indispensable feature of the swarm in adulthood. As mentioned in the introduction, exploration and exploitation are the two tendencies which are performed by the grasshoppers naturally.

A mathematical model is developed which mimics the behavior of grasshopper swarms and their social interaction. This model is used to design the nature-inspired optimization algorithm known as Grasshopper Optimization Algorithm(GOA). It is a population-based algorithm. GOA can be used to solve optimization problems in different fields. It can accurately predict data clusterings. It can also be used to solve real problems containing unknown search spaces.

Each grasshopper in the swarm depicts a solution in the population. Its position is determined with the help of three forces- social interaction ((S_i)), gravity forces (G_i) and the wind direction ((A_i)). The final form of the three affected forces on each grasshopper can be represented as [23] :

$$X_i=S_i+G_i+A_i \tag{34}$$

We can also represent its random behavior as:

$$X_i=r_1S_i+r_2G+r_3A_i \tag{35}$$

Where r_1,r_2 and r_3 are the random values with range [0,1]. The social interaction force among the grasshoppers can be represented as shown below [26]:

$$S_i=\sum_{j=1}^{N} s(d_{ij}).\widehat{d_{ij}},\ j\neq i \tag{36}$$

Received: 22 Sep 2019 | Revised: 13 Oct 2019 | Accepted: 15 Jan 2020

Where d_{ij} denotes the distance between i^{th} and j^{th} grasshopper as shown below:

$$d_{ij} = |x_j - x_i| \tag{37}$$

$\widehat{d_{ij}}$ is a unit vector from the i^{th} grasshopper to the j^{th} grasshopper which can be defined as,

$$\widehat{d_{ij}} = \frac{x_j - x_i}{d_{ij}} \tag{38}$$

S is a function that signifies the strength of two social forces i.e. attraction and repulsion between grasshoppers. It can be defined as:

$$s(r) = fe^{\frac{-r}{l}} - e^{-r} \tag{39}$$

Where f,l are the intensity of the attraction and the attraction length scale. The social behavior is influenced by the changing of the parameters f and l. The distances are in the range of [0,15]. The range of repulsion is (0, 2.079). The secure distance of a grasshopper is 2.079 units from another grasshopper, in which it neither come up against attraction or repulsion. Hence, this area is named as comfort zone. When the distance grow from 2.079, the attraction force increases in the beginning,and then it starts to decrease as it reaches 4. When the distance between the grasshoppers becomes greater than 10, it fails to obey with the function ‘s’. This function doesn’t function on large distances. In order to avoid this problem the distance is always retained in between 1 and 4 [26]. The second force influencing the position of the grasshopper is the gravity force which is calculated as mentioned below:

$$G_i = -g\widehat{e_g} \tag{40}$$

Where g is the gravitational constant and $\widehat{e_g}$ is the center of earth unity vector. As mentioned above, the nymphs do not possess wings due to which they are easily swayed by the wind direction. The wind direction can be calculated as mentioned below:

$$A_i = u\widehat{e_w}; \tag{41}$$

Where u represents a constant and $\widehat{e_w}$ denotes the wind direction unity vector. The position of the grasshopper after replacing the values is:

$$X_i = \sum_{j=1}^{N} s(d_{ij})\widehat{d_{ij}} - g\widehat{e_g} + u\widehat{e_w} \tag{42}$$

ISSN: 1475-7192

The grasshoppers are prevented to obtain the comfort zone rapidly and the swarm is required to be converged to a particular point with the purpose to solve the optimization problem. The equation is therefore, altered to:

$$X_i^d = \left(\sum_{j=1}^{N} c \frac{ub_d - lb_d}{2} s\left(\left| x_j^d - x_i^d \right| \right) \frac{x_j - x_i}{d_{ij}} \right) + \overline{T}_d \tag{43}$$

Where ub_d and lb_d are the upper and lower bound in the D^{th} dimension respectively. C is well known as the decreasing coefficient to lessen the comfort zone, repulsion zone, and attraction zone. Exploitation is advanced as the number of iteration increases. The comfort zone is reduced by c proportional to iteration count and is calculated as follows [23]:

$$C = c_{max} - l \frac{c_{max} - c_{min}}{L} \tag{44}$$

Where C_{max} is the maximum value, C_{min} is the minimum value, l indicates the current iteration and L is the maximum number of iterations [22]. This mathematical model helps the grasshopper to move gradually. In GOA, it is presumed that the grasshopper with best objective value is fittest grasshopper in the course of optimization.

- <u>Advantages</u>

1. This algorithm is simple.
2. It provides gradient-free mechanism.
3. Contains high local optima deflection.
4. Widely used in science and industry.

I. Ant Lion Optimization Algorithm (ALO)

ALO is a recently developed stochastic search algorithm. It imitates the hunting method of antlions in nature. In this new approach, search agents are the ants and antlions and are proposed to find solutions by steps of hunting prey, which includes the random walk of ants, building traps, entrapping ants, catching prey, and reconstructing traps.

Ant lions drill holes which are in the shape of cones in the sand and move around in a circular track to hunt their prey. They wait for the prey inside the holes. There is a high probability of seizing a prey in bigger holes and are mostly created by a fitter elite ant lion. Once an insect is caught, Ant lions draw it under the soil by throwing the sand towards the outer edge of the hole using its big jaw, which doesn't let the insect get away. Then, ant lions eat the prey, throw the remnants outside the hole, rebuild the hole, and get ready for the next hunt [41].

The location of ants and corresponding fitness function matrix are represented below:

$$M_{ant} = \begin{bmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,d} \\ A_{2,1} & A_{2,2} & \cdots & A_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \cdots & A_{n,d} \end{bmatrix} \tag{45}$$

$$M_{antlion} = \begin{bmatrix} AL_{1,1} & AL_{1,2} & \cdots & AL_{1,d} \\ AL_{2,1} & AL_{2,2} & \cdots & AL_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ AL_{n,1} & AL_{n,2} & \cdots & AL_{n,d} \end{bmatrix} \quad (46)$$

The n and d denote the ant's /ant lion's number and their dimension respectively. Ai,j and ALi,j signifies the value of jth dimension of ith ant/ant lion respectively. Each row in M_{ant} represents a solution to a given optimization problem based on 'd' decision variables. The ants/ant lions are computed using objective function f and their evaluation values are stored in M_{OA} and M_{OAL} matrices [41]..

$$M_{OA} = \begin{bmatrix} f(A_{1,1}, A_{1,2}, \cdots, A_{1,d}) \\ f(A_{2,1}, A_{2,2}, \cdots, A_{2,d}) \\ \cdots, \cdots, \cdots, \cdots \\ f(A_{n,1}, A_{n,2}, \cdots, A_{n,d}) \end{bmatrix} \quad (47)$$

$$M_{OAL} = \begin{bmatrix} f(AL_{1,1}, AL_{1,2}, \cdots, AL_{1,d}) \\ f(AL_{2,1}, AL_{2,2}, \cdots, AL_{2,d}) \\ \cdots, \cdots, \cdots, \cdots \\ f(AL_{n,1}, AL_{n,2}, \cdots, AL_{n,d}) \end{bmatrix} \quad (48)$$

The following operations are performed in this algorithm:

1. Construction of Traps: It is assumed that every ant is going to be caught by one ant lion. An ant lion is picked based on its hunting ability that exists in the objective function. The fitter ant lions have more ability to catch prey [42].

2. Generating ant's random walks: Naturally, ants walk in random paths searching for their food. Ants move stochastically in nature when seeking food, therefore, a random walk for an ant at every step of optimization process is shown below:

$$X_i^n = [0, \text{cumsum}(2r(n_1)-1); \text{cumsum}(2r(n_2)-1); \ldots \text{cumsum}(2r(n_N)-1)] \quad (49)$$

where n denotes the maximum number of iterations cumsum represents computing cumulative sum, and 't' represents step of random walk.

$$r(t) = \begin{cases} 1 & \text{if rand} > 0.5 \\ 0 & \text{if rand} < 0.5 \end{cases}$$

(50)

where *rand* represents random number generator which has the range in [0,1]. With the help of the below given equation, random walk can be made within the search space:

$$X_i^n = \frac{(X_i^n - a_i) \times (d_i - c_i^n)}{(b_i^n - a_i)} + c_i \quad (51)$$

3. Trapping in ant lion pits: The mathematical equations for trapping is given by:

$$c_i^n = c^n + antlions_j^n \quad (52)$$

$$d_i^n = d^n + antlions_j^n \quad (53)$$

Where c_i^n and d_i^n denote the minimum and maximum of all variables at n^{th} iteration. c^n and d^n are minimum and maximum of all variables for i^{th} ant.

4. Sliding trapped ants: Ant lions push ants towards their constructed traps to prevent them from fleeing. This process is mathematically developed by iteratively lowering the lower and upper bounds of ant's random walk as shown below:

$$c^n = \frac{c^n}{I} \quad (54)$$

$$d^n = \frac{d^n}{I} \quad (55)$$

where: c^n and d^n denote the minimum and the maximum of all variables at nth iteration respectively, and 'i' is defined as follows:

$$I = 10^w \frac{n}{N} \quad (56)$$

where 'w' signifies a dynamic parameter used to control the search exploitation level.

5. Catching ants and reconstruction of the hole: The final step of hunting process is to ingest the trapped ants and re-constructing the previously used hole. Mathematically, this process is developed by comparing the fitness of eaten ants with their hunting ant lions. If an ant becomes more fitter than its corresponding ant lions, it signifies that it is about to be eaten. Accordingly, Ant lions substitute their positions with the eaten ants so as to improve their chances for the next attack. It is represented as shown:

$$Antlions_j^n = Ant_i^n \quad \text{If } f(Ant_i^n) > f(Antlions_j^n) \quad (57)$$

6. Elitism: So as to preserve the best obtained solution the algorithm adopts an elitism strategy across different iterations which is as follows:

$$\mathrm{Ant}_i^n = \frac{(R_A^n + R_E^n)}{2}$$

(58)

where R_A^n denotes random wander around the selected ant lion, and R_E^n represents the random walk around the elite ant lion at the nth iteration.

➢ Advantages:

1. High speed.
2. Good Performance.
3. Use of few parameters.

➢ Limitations:

1. prematurity and local optimum may arise in complex problems.

➢ Applications:

1. multi-objective problems in engineering design.
2. process planning and scheduling.
3. Text classification.
4. Bioinformatics.

J. *Crow Search Algorithm (CSA)*

Crows are known to be very intelligent creatures. It is widely known that crows can retain information for a very long time even upto several months.. They have large heads in contrast to their body size. They live in groups. They can hide the excess food in hiding and retrieve it when needed. Also, they have the ability to observe other crows to know about their hiding places so that they can steal their food. If the crow feels another crow is following it, it travels to a place far away from the hiding place to fool the thief. Based on this social behavior of crows a metaheuristics optimization algorithm is developed which is widely known as Crow Search Algorithm(CSA). CSA is a population based method. It mimics the crow's behavior.

In CSA, the search space depicts the environment and the crows in the group depict the search agent(solution) in the population. It is assumed that the population contains N solution. The position of each crow i at iteration t is represented by a vector x_i^t, where $x_i^t = [x_{i1}^t, x_{i2}^t, \ldots, x_{id}^t]$ and d is the problem dimension [27]. Each crow (individual) is assumed to have the capability to memorize the best visited location $M_{i,k}$ to hide food until the current iteration, where

$$M_{i,k} = [M_{i,k}^1, M_{i,k}^2, \ldots, M_{i,k}^n] \qquad (59)$$

The position of both the values are updated according to two behaviors: Pursuit and evasion.

- **Pursuit:** A crow j follows crow i with the purpose to determine its hidden place. The crow i overlooks the presence of the other crow, thereby, the motive of the crow j is achieved.
- **Evasion**: The c8row i is aware of the presence of crow j and so as to shield its food, crow i deliberately take a random flight.

This behavior is simulated in CSA through the implementation of a random movement. The sort of behavior considered by each crow i is determined by an awareness probability (AP). Hence, a random value ri uniformly distributed between 0 and 1 is sampled. If ri is greater or equal than AP, behavior 1 is applied, otherwise scenario two is chosen. This operation can be summarized in the following model [27]:

$$x_{i,k+1} = \{x_{i,k} + r_i \cdot f\, l_{i,k} \cdot (M_{j,k} - x_{i,k}), \text{if } r_i \geq AP\} \quad (60)$$
$$\{\text{Move to random position, otherwise}\}$$

The flight length f $l_{i,k}$ parameter represent the magnitude of movement from crow $x_{i,k}$ towards the best position$M_{j,k}$ of crow j, r_i is a random number with uniform distribution in the range [0, 1]. The value of AP is adjusted according to the exploration and exploitation processes. For performing global search the value of AP is increased and for local search, its value is decreased. Local search means that the crows would intensify their search in a small area which would increase the probability of finding the hideout. In global search the crows fly around in random manner in the whole search space. This would likely lower down the probability of finding the real food hideouts [27]. Each crow updates its value on the basis of the fitness function. The new position is only preferred if the fitness value of the crow's new position is better than the current memory's value. Otherwise the value is not changed. If their position is modified then the memory vector is updated as follows [28]:

$$M_{i,k+1} = \{F(x_{i,k+1})\ ;\ F(x_{i,k+1}) < F(M_{i,k})\} \quad (61)$$

$$\{M_i\ ;\ \text{k otherwise}\}$$

VIII. FEATURE SELECTION USING BIO-INSPIRED ALGORITHMS

When the search space is large, feature subset selection becomes tough and hard to manage. Bio-inspired algorithms are perfect tools for such problems as they can find a quality solution within a reasonable amount of time [36]. Some of the objectives of feature subset selection are to improve accuracy, decrease the number of features and good data understanding [2]. Top reasons to use feature selection are:

- Machine learning algorithms can be trained faster.
- The problem can be easily understood.
- An appropriate feature subset can improve accuracy.
- Overfitting can be reduced [5].

Bio-inspired algorithms are based on the life of living organisms and creatures. It has picked up many qualities like gene evolution, insect swarming, bird swarming or food foraging. Many kinds of research have been made on the bio-inspired algorithms over the past years. A fitness value is the most important factor which helps in the bio-inspired algorithm. [36].

IX. BIO-INSPIRED ALGORITHMS IN BIOMEDICAL FIELD

Computer science and biology have had a long-lasting and healthy relationship for decades. Biologists depend on computational techniques to inspect and integrate large data sets, while various computational methods were inspired by the high level design principles of biological systems.These two methods are found to be coinciding lately [40]. BioMedical data analyses are steering new optimization research trends mainly established on machine learning and artificial intelligence, motivating intersections with biomedical imaging & data analysis and systems development. Bio-inspired computing aims to apply significant information-processing aptitudes of the natural realm to the computation realm. It demonstrated a strong link with computational biology and other biology-inspired computing models as well because of its effectiveness and uniqueness.

A. Crow Search Algorithm implementation and Results

Other bio inspired algorithms were also tested out of which crow search algorithm was picked for this study. Crow Search Algorithm picks the important features before training the model. The models are then trained and tested using the classification algorithms. The goal of classification is to predict the target class for each case in data. In this study, two machine learning algorithms- KNN and SVM were applied to have a broader understanding of the healthcare dataset having 10 exploratory variables. The results are summarized in Table I.

TABLE I: Result

Algorithms	Accuracy	Features Selected
KNN	0.943	5
SVM	0.989	4

We can conclude from the above table that the liver disease can be more accurately predicted using the SVM algorithm as it has a better accuracy of 98%.

X. CONCLUSION

In this work, various bio-inspired algorithms were studied. These are known to be the most powerful algorithms to solve complex optimization problems. These are still evolving and studied rigorously by the scientists. This paper focused on how the algorithms work and how the algorithms can be applied on the healthcare dataset.

REFERENCES

1. Madhu Sadhvi (Sep 26, 2017), Chapter 1: Complete Linear Regression with Math, Medium. www.medium.com/deep-math-machine-learning-ai/
2. Reena Shaw (June 26, 2019), The 10 Best Machine Learning Algorithms for Data Science Beginners,Dataquest.https://www.dataquest.io/blog/top-10-machine-learning-algorithms-for-beginners/

Received: 22 Sep 2019 | Revised: 13 Oct 2019 | Accepted: 15 Jan 2020

3. Adel Sabry Eesa, Adnan Mohsin Abdulazeez Brifcani, Zeynep Orman , "Cuttlefish Algorithm – A Novel Bio-Inspired Optimization Algorithm",International Journal of Scientific & Engineering Research, Volume 4, Issue 9, September 20
4. Problems Cuttlefish Algorithm", International Journal of Computer and Information Engineering Vol:8, No:9, 2014
5. Saurav Kaushik (December 1, 2016), Introduction to Feature Selection methods with an example, Analytics vidhya. https://www.analyticsvidhya.com/blog/2016/12/introduction-to-feature-selection-methods-with-an-example-or-how-to-select-the-right-variables/
6. M.P. Saka, Ibrahim Aydogdu "Analysis of Swarm Intelligence–Based Algorithms for Constrained Optimization", Swarm Intelligence and Bio-Inspired Computation, 2013
7. Xin-She Yang, Mehmet Karamanoglu,'Swarm Intelligence and Bio-Inspired Computation', Swarm Intelligence and Bio-Inspired Computation, 2013.
8. Amir H. Gandomi et al, ' Metaheuristic Algorithms in Modeling and Optimization', Metaheuristic Applications in Structures and Infrastructures, pp.1-24.
9. Shu-Chuan Chu et al, 'Cat swarm optimization", Conference: PRICAI 2006: Trends in Artificial Intelligence, August 7-11, 2006.
10. Fran Sérgio Lobato & Valder Steffen Jr, "Fish swarm optimization algorithm applied to engineering system design", Latin American Journal of Solids and Structures, vol.11 no.1 Jan.2014.
11. Mohit Kumar et al, "Optimization of Some Standard Functions using Artificial Algae Algorithm", International Journal of Engineering Research & Technology (IJERT), ACMEE - 2016 Conference Proceedings.
12. Md.Al-imran Roton (May 26, 2016), Bat algorithm and applications, Slideshare. https://www.slideshare.net/roton007/bat-algorithm-and-application
13. Bhavana Bansal & Anita Sahoo, "Bat Algorithm for Full Model Selection in Classification: A Multi-objective Approach", IJIACS,Volume 4,March 2015 .
14. Primitivo Díaz et al, "An Improved Crow Search Algorithm Applied to Energy Problems", March 2018
15. "Artificial bee colony algorithm", Wikimedia Foundation, 12 Jan 2020, https://en.wikipedia.org/wiki/Artificial_bee_colony_algorithm
16. Kamalpreet Kaur Dhaliwal et al, "Modified Cat Swarm Optimization Algorithm for Design and Optimization of IIR BS Filter", (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (3), 2015
17. A.Y. Abdelaziz, A. Fathy, "A novel approach based on crow search algorithm for optimal selection of conductor size in radial distribution networks", Engineering Science and Technology, Journal 20 (2017) 391402
18. Askarzadeh, A., "A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm", Computers & Structures, Volume 169, June 2016, Pages 1-12.
19. Hala M.Alshamlan, Ghada H.Badr, Yousef A.Alohali, "Genetic Bee Colony (GBC) algorithm: A new gene selection method for microarray cancer classification", Computational Biology and Chemistry, Volume 56, June 2015, Pages 49-60
20. N. Vinayak (Jun 2010, 2015), ABC Algorithm - Basic Algorithm on Optimization, Slideshare. https://www.slideshare.net/VinayakNayak5/abc-algorithm-49203658

21. Shahrzad Saremi, Seyedali Mirjalili, Andrew Lewis, "Grasshopper Optimisation Algorithm: Theory and application", Advances in Engineering Software 105:30-47 · March 2017.
22. Neve et al., "Application of Grasshopper Optimization Algorithm for Constrained and Unconstrained Test Functions", International Journal of Swarm Intelligence and Evolutionary Computation, 2017.
23. Xin-She Yang,"Nature-Inspired Algorithms and Applied Optimization Studies", Computational Intelligence, 2018.
24. A. A. Abou El Ela ; Ragab A. El-Sehiemy ; A. M. Shaheen ; A. S. Shalaby, "Application of the crow search algorithm for economic environmental dispatch", 2017 Nineteenth International Middle East Power Systems Conference (MEPCON)
25. Ahmed Fouad Ali (Apr 20, 2018), Grasshopper optimization algorithm, slideshare. https://www.slideshare.net/afar1111/grasshopper-optimization-algorithm
26. Ahmed Fouad Ali (May 13, 2019), Crow Search Algorithm, Slideshare. https://www.slideshare.net/afar1111/crow-search-algorithm?qid=80054cf7-2297-400d-a60a-6ff013736fea&v=&b=&from_search=1.
27. Almoataz Y. Abdelaziz, Ahmed Fathy, "A novel approach based on crow search algorithm for optimal selection of conductor size in radial distribution networks",Engineering Science and Technology, an International Journal, Volume 20, Issue 2, April 2017, Pages 391-402.
28. Budi Santosa ; Mirsa Kencana Ningrum, "Cat Swarm Optimization for Clustering", IEEE, 2009 International Conference of Soft Computing and Pattern Recognition.
29. Pawn Bhambu, "Artificial Bee Colony Algorithm: A Survey", International Journal of Computer Applications 149(4):11-19 · September 2016.
30. Ryo Takano, Tomohiro Harada, Hiroyuki Sato, Keiki Takadama, "Artificial Bee Colony Algorithm Based on Local Information Sharing in Dynamic Environment", Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems, Volume 1 pp 627-641.
31. Yunfeng Xu, Ping Fan and Ling Yuan, "A Simple and Efficient Artificial Bee Colony Algorithm",Mathematical Problems in Engineering Volume 2013.
32. Lijun Sun, Tianfei Chen and Qiuwen Zhang, "An Artificial Bee Colony Algorithm with Random Location Updating", Scientific Programming, Volume 2018, Article ID 2767546, 9 pages.
33. Fran Sérgio Lobato, Valder Steffen, Jr, "Fish Swarm Optimization Algorithm Applied to Engineering System Design", Latin American Journal of Solids and Structures 11(1):143-156 · January 2014.
34. Ashraf Darwish, "Bio-inspired computing: Algorithms review, deep analysis, and the scope of applications", Future Computing and Informatics Journal 3, 2018.
35. Chulmin Yun et al, "Feature Subset Selection Based on Bio-Inspired Algorithms", Journal of Information Science And Engineering, 1667-1686 (2011).
36. E.Emarya, Hossam M.Zawbaa, Aboul Ella Hassanien, "Binary grey wolf optimization approaches for feature selection ", Neurocomputing, Volume 172, 8 January 2016, Pages 371-381.
37. Shilpa Kukreja, Surjeet Dalal, "Nature-Inspired Algorithms: Critical Study", International Journal of Recent Research Aspects, 2016, pp. 38-40.
38. V.R. Balasaraswathi et al, " Chaotic CuttleFish Algorithm for Feature Selection of Intrusion Detection System", International Journal of Pure and Applied Mathematics Volume 119 No. 10 2018, 921-935.

39. Ms. Neeta Nemade , Mr. Dhiraj Rane, "A Review on Bio-Inspired Computing Algorithms and Application", IOSR Journal of Computer Engineering (IOSR-JCE), PP 12-19.
40. Eslam Hamouda et al, "Ant Lion Optimization algorithm for kidney exchanges", PLoS One, May 2018.
41. Mirjalili S. The Ant Lion Optimizer. Advances in Engineering Software. 2015;83(2015):80–98.
42. Shallu Sehgal, Dr Manisha Agarwal, Dr Deepak Gupta**,** Arun Bashambu "Comparative Study on Machine Learning and Data Mining Techniques for Diseases Diagnosis", International Journal of Management, IT and Engineering Vol. 9 Issue 1(1), January 2019, ISSN: 2249-0558.
43. Shallu Sehgal, Manisha Agarwal**,** "Analogous Examination of Various Machine Learning Algorithm Applied to Big Data", IEEE International Conference on Advances in Computing, Communication Control and Networking (ICACCC-2018 Co -Sponsored by IEEE UP Section), October 2018.