Appl Intell
DOI 10.1007/s10489-016-0767-1

Training feedforward neural networks using multi-verse optimizer for binary classification problems

Hossam Faris[1] · Ibrahim Aljarah[1] · Seyedali Mirjalili[2]

Abstract This paper employs the recently proposed nature-inspired algorithm called Multi-Verse Optimizer (MVO) for training the Multi-layer Perceptron (MLP) neural network. The new training approach is benchmarked and evaluated using nine different bio-medical datasets selected from the UCI machine learning repository. The results are compared to five classical and recent evolutionary metaheuristic algorithms: Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Differential Evolution (DE), FireFly (FF) Algorithm and Cuckoo Search (CS). In addition, the results are compared with two well-regarded conventional gradient-based training methods: the conventional Back-Propagation (BP) and the Levenberg-Marquardt (LM) algorithms. The comparative study demonstrates that MVO is very competitive and outperforms other training algorithms in the majority of datasets in terms of improved local optima avoidance and convergence speed.

Keywords Multi-verse optimizer · MVO · Multilayer perceptron · MLP · Training neural network · Evolutionary algorithm

✉ Hossam Faris
hossam.faris@ju.edu.jo

Ibrahim Aljarah
i.aljarah@ju.edu.jo

Seyedali Mirjalili
seyedali.mirjalili@griffithuni.edu.au

[1] Business Information Technology Department, King Abdullah II School for Information Technology, The University of Jordan, Amman, Jordan

[2] School of Information and Communication Technology, Griffith University, Nathan, Brisbane, QLD 4111, Australia

1 Introduction

Artificial Neural Networks (ANNs) are mathematical models widely utilized for modeling complex nonlinear processes. Some of the attractive characteristics of ANNs include the ability to capture nonlinearity, they are highly parallel and their fault/noise tolerance [1]. One of the most popular neural network is the feedforward Multi-layer Perceptron (MLP) [23]. MLP has been successfully applied to data classification, pattern recognition and function approximation problems as the literature shows.

One of the most important and key aspects of neural networks is the training process. The goal of an MLP trainer is to find the best set of weights that minimizes the prediction or classification error. In general, training algorithms can be classified into two groups: gradient-based algorithms versus stochastic search algorithms. Gradient based algorithms are mostly conventional and mathematical optimizers. In the literature, Back Propagation (BP) algorithm [24] and its variants are the most widely applied and used gradient-based training algorithms. Although such algorithms benefit from fast convergence speed, they suffer some disadvantages such as high dependency to the initial solution, which might have a substantial negative impacts on the convergence of the algorithm as well, and local optima entrapment, which is a common issue in complex nonlinear problems [27].

On the other hand, stochastic search methods like metaheuristic algorithms were proposed by researchers as alternatives to gradient-based methods for training MLP networks. Metaheuristic algorithms proved to be more efficient in finding a global solution when the search space is challenging, large, and little information is known about the problem itself. Metaheuristics such as Genetic Algorithms

Published online: 05 March 2016

(GA), Particle Swarm Optimization (PSO) and their variants are the most well-known nature inspired MLP trainers [2, 6, 16, 21, 25, 26, 35].

GA is an evolutionary algorithm inspired by the Darwinians' theory of evolution and natural selection [7]. Montana and Davis proposed one of the earliest works on training MLP networks with GA [21]. They showed that GA is able to outperform BP when solving real and challenging problems. Mendes et al. [16] showed that PSO, which is an algorithm inspired by bird flocks, could be efficient in cases of a large number of local minima. In 2008, Slowik and Bialko [28] employed another evolutionary algorithm called Differential Evolution (DE) for training MLP and showed that it has promising performance compared to gradient methods like BP and Levenberg-Marquardt methods. Other recent nature-inspired metaheuristic algorithms applied for training MLP networks are: Krill Herd (KH) [10, 11], Cuckoo Search (CS) [29], Firefly [22], Grey Wolf Optimizer (GWO) [17], Gravitational Search Algorithm (GSA) [18] and Biogeography-Based Optimizer (BBO) [19]. MLP networks trained by metaheuristic algorithms have shown promising results in different specific domain problems like multispectral image classification [14], e-Learning applications [9], bio-medical applications [33] and industrial processes [2].

Despite the merits of the above-mentioned recent trainers, there is a question here as if we still need to design new training algorithms. There answer is yes according to the No-Free-Lunch (NFL) theorem [32]. This theorem logically has proven that there is no optimization technique for solving all optimization problems. Training MLP is also an optimization problem that changes for every dataset. Therefore, the current optimizers have the potential to fail training MLP for current or new problems effectively. This is the motivation of this work, in which we use the recently developed Multi-Verse Optimizer (MVO) for training MLP networks. The contributions are as follows:

1. A novel trainer based on the MVO algorithm is proposed.
2. Nine datasets are solved by the proposed trainer.
3. The application of the trainer is investigated in bio-medical field.

MVO is a new metaheuristic evolutionary algorithm proposed in [20]. MVO is inspired from multi-verse theory in physics. High local optima avoidance and fast convergence speed of MVO were our primary motivations to chose this algorithm. The efficiency and performance of the proposed MVO training method are evaluated and tested based on nine different challenging bio-medical datasets. In addition, the results are compared with popular and recent training methods like BP, LM, GA, PSO, DE, Firefly and Cuckoo Search.

The remainder of the article is organized as follows: Section 2 gives a brief description of MLP neural networks. Section 3 represents the MVO metaheuristic algorithm and discusses its main characteristics. In Section 4, we discuss the new MVO-based trainer and show how it can be used for training MLP networks. The experiments and results are presented and analyzed in Section 5. Finally, the findings of this work are concluded in Section 6.

2 Feedforward multilayer perceptron neural networks

Artificial Neural Networks (ANNs) are mathematical models and universal approximators inspired by the function of biological nervous systems [5, 8]. ANNs consist of a set of processing elements called "neurons". Feedforward multilayer perceptron neural network (FFNN) is one of the most common types of ANNs. In FFNN, the neurons are arranged in layers and fully interconnected to form a directed graph. The layers of the FFNN are the input layer, a number of hidden layers, and the output layer. An example of a simple FFNN with a single hidden layer is shown in Fig. 1. Connections between neurons are represented as weights. Each neuron consists of a summation function and an activation function. Summation functions sums up the product of inputs and weights, and a bias as shown in (1) where w_{ij} is the connection weight connecting I_i to neuron j, β_j is a bias weight and n is the total number of neuron inputs.

$$S_j = \sum_{i=1}^{n} w_{ij} I_i + \beta_j \tag{1}$$

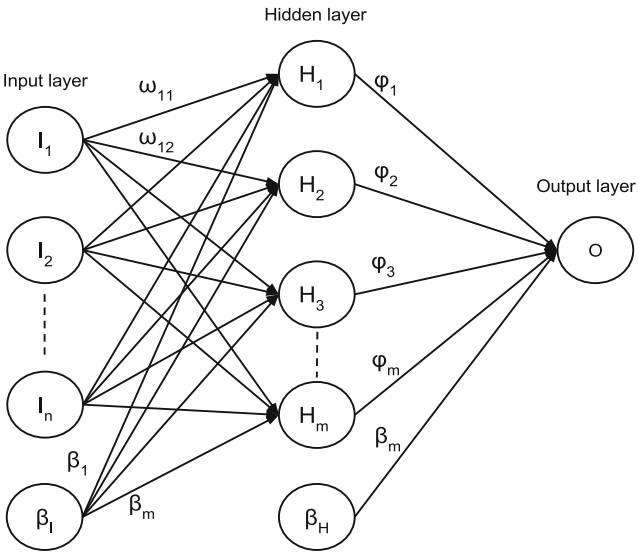

Fig. 1 Simple Artificial Neural Network architecture

The output of the summation function will be an input to an activation function (also called transfer function). Usually, a nonlinear activation function like the S-shaped curved sigmoid function is used. Sigmoid function is shown in (2).

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2}$$

Therefore, the output of the neuron j can be described as in (3).

$$y_j = f_j \left(\sum_{i=1}^{n} w_{ij} I_i + \beta_j \right) \tag{3}$$

After constructing the neural network, the set of weights of the network are tuned to approximate the needed results. This process is carried out by applying a training algorithm to adjust the weights until some error criteria is met.

3 Multi-verse optimizer

MVO is a recently proposed Evolutionary Algorithm (EA) [20]. Similarly to other EAs, it starts the optimization process by creating a population of solutions and improves it over a predefined number of generations. The improvement of individuals in each population is done based on one of the theories about the presence of multiple universes. In fact, this algorithm mimics the interaction between multiple universes through white hole, black hole, and worm hole.

As can be seen in Fig. 2, in MVO each solution is considered as a universe and each object in the universe is assumed as a variable of a given problem. For instance, a problem with 5 variables will have universes with 5 objects. The main idea of this algorithm originates from the fact that larger universes tend to send objectives to smaller universes to reach stable status. A large universe is defined based on inflammation rate in the multi-verse theory. During optimization, this algorithm follows the following main rules:

- The presence of a white hole is proportional to the inflation rate
- The presence of a black hole is inversely proportional to the inflation rate
- Objects move from a white hole to a black hole
- The objects of all universes may be replaced by the objects of the universe with the highest inflammation rate

The overall process of the MVO algorithm is illustrated in Fig. 3. It can be evidently seen in this figure that a universe with a lower inflammation rate tend to accept more object from better universes as well as the best universe formed so far to improve its inflammation rate.

The mathematical formulation of this algorithm is as follows:

$$x_i{}^j = \begin{cases} \begin{cases} x_j + TDR + ((ub_j - lb_j) * r_4 + lb_j)) & \text{if} r_3 < 0.5 \\ x_j - TDR + ((ub_j - lb_j) * r_4 + lb_j)) & \text{if} r_3 \geq 0.5 \end{cases} & \text{if} \quad r_2 < WEP \\ x_i^j & \text{if} \quad r_2 \geq WEP \end{cases} \tag{4}$$

where X_j shows the jth variable in the bests universe, TDP/WEP are coefficients, lb_i shows the lower bound in jth variable, ub_i shows the upper bound in jth variable, r_2, r_3, r_4 are random numbers in the interval of [0, 1], and x_i^j indicates the jth parameter in ith universe.

It has been proven that this algorithm is able to outperform other algorithms in the literature on the test functions. It was observed and confirmed that this algorithm benefits from high exploration and exploitation [20]. These motivated our attempts to propose a novel training algorithm based on this algorithm for the first time in the following section.

4 MVO for training MLP

In order use the MVO algorithm for training the MLP network, two important points should be addressed: the representation and design of (MVO) individuals and the formulation of the fitness function (cost function).

In this work, the MVO algorithm is used to train an MLP with single hidden layer. Therefore, each universe in MVO is formed by three parts: the connection weights between the input layer and the hidden layer ω_{ij}, the weights between the hidden layer and the output layer φ_k, and the bias weights β_l. In our implementation, MVO universes are encoded as

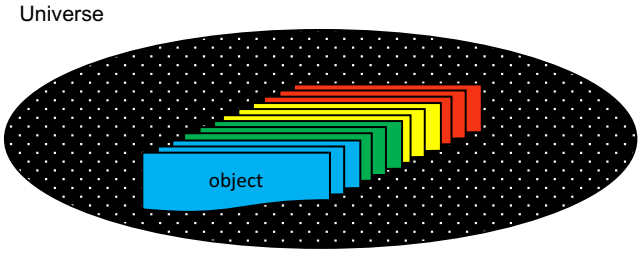

Fig. 2 Concept of universe and object in MVO

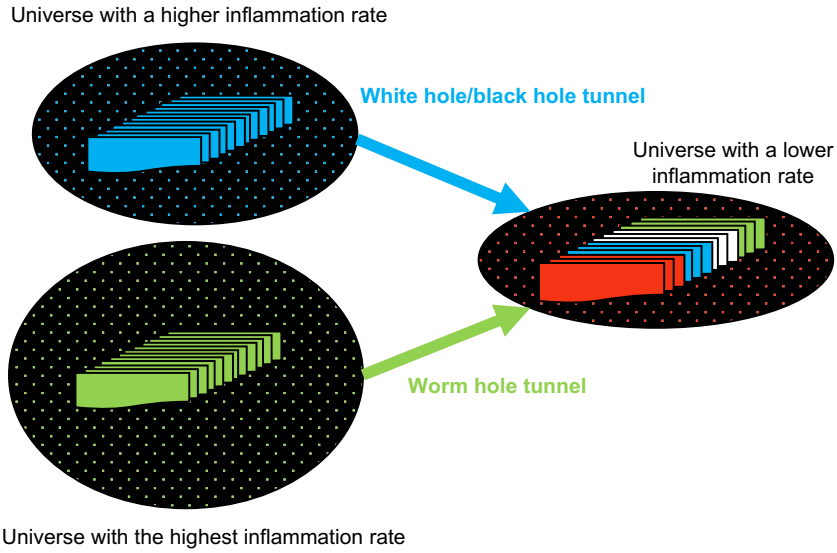

Fig. 3 The overall process of the MVO algorithm

vectors as illustrated in Fig. 4 where vectors are sequence of real numbers each of which belongs to the interval $[-1, 1]$. The number of objects in each universe is given by (5), where n is the number of input features and m is the number of neurons in the hidden layer.

$$IndividualLength = (n \times m) + (2 \times m) + 1 \tag{5}$$

In MVO algorithm every universe is evaluated according to status of its objects. This evaluation is done by passing the vector of weights and biases to the MLP neural network then the Mean Squared Error (MSE) criteria is calculated based on the prediction of the neural network using the training dataset. The MSE criteria is given in (6) where y and $\hat{y}$ are the actual and the estimated values based on pro posed model and n is the number of samples in the training dataset.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y - \hat{y})^2 \tag{6}$$

5 Experiments and results

This section presents a comprehensive analysis to investigate the efficiency of the MVO algorithm for training MLP neural networks. The results obtained by MVO approach is evaluated on nine well-known datasets to conduct a reliable comparison. We present the comparison of the MVO with five well-known metaheuristic algorithms: GA [21], PSO [16], DE [28], FF [22] and CS [29] which have been used in the literature to train the MLP neural network. In addition, we compare our results with the BP and LM techniques, which are considered the most common gradient based methods for training the MLP neural network.

5.1 Experimental setup

The experiments were conducted on a personal computer with 4GB of RAM and, 4 Intel cores (2.67GHz each). For all experiments, we used Matlab R2010b to implement the proposed MVO technique and other algorithms. All datasets are divided into 66 % for training, and 34 % for testing. All experiments are executed for 10 different runs and each run includes 200 iterations. Furthermore, we used the common parameter settings for MVO, GA, PSO, DE, FF and CS that are recommended in literature as shown in Table 1.

According to literature, there is no standard method for selecting the number of neurons in the hidden layer. In this work, we follow the method proposed in [17, 19, 30] where

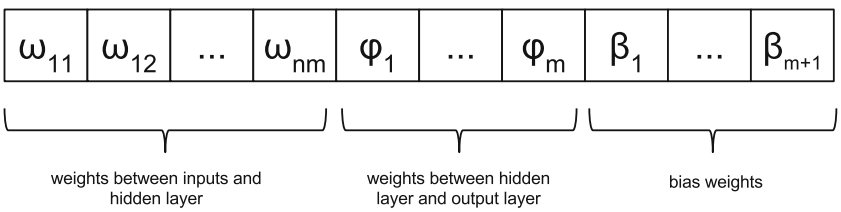

Fig. 4 Representation of MVO individuals structure

Table 1 The initial parameters of the metaheuristic algorithms

Algorithm	Parameter	Value
GA	• Crossover probability	0.9
	• Mutation probability	0.1
	• Selection mechanism	Stochastic Universal Sampling
	• Population size	50
	• Number of generations	200
PSO	• Acceleration constants	[2.1,2.1]
	• Intertia weights	[0.9,0.6]
	• Number of particles	50
	• Number of generations	200
DE	• Crossover probability	0.9
	• Differential weight	0.5
	• Population size	50
	• Number of generations	200
FF	• Alpha	0.2
	• Beta	1
	• Gamma	1
	• Number of fireflies	50
	• Number of generations	200
Cuckoo Search	• Discovery rate P_α	0.25
	• Number of nests	50
	• Number of generations	200
MVO	• Minimum wormhole existence probability	0.2
	• Maximum wormhole existence probability	1
	• Number of search agents	50
	• Number of generations	200

the number of hidden neurons equals to $2 \times N + 1$ where N is number of features in the dataset.

All input features are mapped to the interval of [0, 1] to put all features in the same scale. In our experiments, Min-max normalization is applied to perform a linear transformation on the original data as given in (7), were v' is the normalized value of v in the range $[min_A, max_A]$.

$$v' = \frac{v_i - min_A}{max_A - min_A} \tag{7}$$

5.2 Binary classification problems

The proposed MVO approach for training MLP networks is evaluated and benchmarked based on nine known real datasets, which are selected from the University of California at Irvine (UCI) Machine Learning Repository [1] [12]. All selected datasets are binary classification problems. Table 2 describes these datasets in terms of number of features, number of training samples and number of testing samples.

Table 2 Summary of the Binary classification datasets

Dataset	#Features	#Training samples	#Testing Samples
Blood	4	493	255
Breast cancer	8	461	238
Diabetes	8	506	262
Hepatitis	10	102	53
Vertebral	6	204	106
Liver	6	79	41
Diagnosis-I	6	79	41
Diagnosis-II	6	79	41
Parkinsons	22	128	67

The details of all datasets are given in the following paragraphs:

- Blood: this dataset is part of the donor database of Blood Transfusion Service Center in Hsin-Chu City in Taiwan. The dataset contains 748 donors instances that were selected randomly from the donor database. The dataset is a binary classification problem with an output class variable representing whether the person donated blood in a time period (1 stand for donating blood; 0 stands for not donating blood). Input variables are Recency: months since last donation, Frequency: total number of donation, Monetary: total blood donated in c.c.), and Time: months since first donation [34].
- Breast cancer: this dataset was originally obtained from from Dr. William H. Wolberg, the University of Wisconsin Hospitals, Madison. It this dataset contains 699 instances where each instance represents a patient that had undergone surgery for breast cancer. Four variables are measured for each patient and labeled as benign or malignant [15, 31].
- Diabetes: this dataset is a part of a larger dataset donated by the National Institute of Diabetes and Digestive and Kidney Diseases. Instances of the dataset represent patients who are are Pima-Indian women at least 21 years old and living near Phoenix, Arizona, USA. The class label of the dataset is binary, '1' for a positive test for diabetes and '0' is a negative test for diabetes. There are 268 (34.9 %) cases identified as positive tests and 500 (65.1 %) cases as negative tests. The measured variables for each case are: 1. Number of times pregnant 2. Plasma glucose concentration a 2 hours in an oral glucose tolerance test 3. Diastolic blood pressure (mm Hg) 4. Triceps skinfold thickness (mm)

[1] http://archive.ics.uci.edu/ml/

Table 3 Accuracy Results

Dataset \ Algorithm		MVO	BP	LM	GA	PSO	DE	FF	CS
Blood	Avg	0.7965	0.7031	0.7616	0.7906	0.7902	0.7753	0.7702	0.7855
	Std	0.0050	0.0191	0.01038	0.0042	0.0042	0.0042	0.0038	0.0056
	Best	0.8000	0.7725	0.7882	0.8000	0.8000	0.7804	0.7765	0.7961
Breast cancer	Avg	0.9731	0.7441	0.9584	0.9723	0.9723	0.9546	0.9723	0.9702
	Std	0.0029	0.2649	0.0104	0.0066	0.0060	0.0151	0.0035	0.0090
	Best	0.9748	0.9454	0.9748	0.9832	0.9790	0.9706	0.9790	0.9832
Diabetes	Avg	0.7679	0.6198	0.7057	0.7641	0.7653	0.6989	0.7595	0.7565
	Std	0.0119	0.0971	0.0244	0.0105	0.0075	0.0353	0.0062	0.0127
	Best	0.7901	0.6908	0.7405	0.7748	0.7748	0.7672	0.7672	0.7748
Habitit	Avg	0.8943	0.6962	0.8547	0.8811	0.8943	0.8453	0.8717	0.8604
	Std	0.0239	0.2145	0.0179	0.0236	0.0221	0.0434	0.0173	0.0428
	Best	0.9434	0.8679	0.8868	0.9057	0.9245	0.9245	0.9057	0.9245
Vertebral	Avg	0.8594	0.6491	0.7906	0.8670	0.8519	0.7783	0.8613	0.8660
	Std	0.0113	0.1704	0.0614	0.0094	0.0046	0.0568	0.0046	0.0132
	Best	0.8774	0.8774	0.8774	0.8774	0.8585	0.8679	0.8679	0.8868
Liver	Avg	0.7246	0.5220	0.6694	0.7169	0.7178	0.5780	0.7059	0.7000
	Std	0.0230	0.0568	0.0304	0.0170	0.0170	0.0411	0.0204	0.0496
	Best	0.7542	0.5847	0.7203	0.7373	0.7542	0.6271	0.7373	0.7881
Diagnosis I	Avg	1.0000	0.8341	1.0000	1.0000	1.0000	0.9366	1.0000	1.0000
	Std	0.0000	0.1384	0.0000	0.0000	0.0000	0.0839	0.0000	0.0000
	Best	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Diagnosis II	Avg	1.0000	0.8512	0.9049	1.0000	1.0000	0.9780	1.0000	1.0000
	Std	0.0000	0.1574	0.2045	0.0000	0.0000	0.0545	0.0000	0.0000
	Best	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Parkinson	Avg	0.9299	0.7731	0.8746	0.8940	0.9179	0.8149	0.8746	0.8627
	Std	0.0434	0.0466	0.0879	0.0382	0.0176	0.0446	0.0246	0.0329
	Best	0.9851	0.8358	0.9552	0.9552	0.9403	0.8657	0.9104	0.9104

5. 2-Hour serum insulin (mu U/ml) 6. Body mass index 7. Diabetes pedigree function 8. Age (years).

– Hepatitis: this dataset was donated by Carnegie-Mellon University. It contains values of 18 features measured for 155 patients affected by Hepatitis. Hepatitis is a type of liver disease. The data was collected to predict if these patients will die or survive. Eighty-five number of the instances are labeled as a DIE class while the remaining 70 are labeled as LIVE class. Eleven features are boolean while the rest are numeric. Five features of the dataset are outcomes of blood tests while the last attribute includes daily alcohol consuming [3].
– Vertebral: this dataset was built by Dr. Henrique da Mota. The dataset consists of values for six biomechanical features used to classify orthopaedic patients into two classes (normal or abnormal). There are 100 normal patients and 210 'abnormal' patients. Abnormal patients are Disk Hernia patients or Spondylolisthesis patients. The six features of the datasets are derived from the shape and orientation of the pelvis and lumbar spine (in this order): pelvic incidence, pelvic tilt, lumbar lordosis angle, sacral slope, pelvic radius and grade of spondylolisthesis. The following convention is used for the class labels: Disk Hernia (DH), Spondylolisthesis (SL), Normal (NO) and Abnormal (AB).
– Liver disorders: the dataset was donated by BUPA Medical Research Ltd. It includes values of 6 features measured for 345 male individuals. Each record in the dataset has a binary label indicating the liver disorder status. There are 5 features collected from blood tests which are sensitive to liver disorders and might arise from excessive alcohol consumption. These

Table 4 MSE Results

Dataset \ Algorithm		MVO	BP	LM	GA	PSO	DE	FF	CS
Blood	Avg	1.52E-01	2.58E-01	1.27E-01	1.53E-01	1.53E-01	1.63E-01	1.56E-01	1.54E-01
	Std	2.82E-04	1.76E-01	9.14E-03	2.53E-04	1.21E-04	5.39E-03	6.87E-04	3.77E-04
	Best	1.52E-01	1.75E-01	1.12E-01	1.52E-01	1.52E-01	1.56E-01	1.55E-01	1.53E-01
Breast cancer	Avg	2.01E-02	2.42E-01	3.90E-03	2.41E-02	2.26E-02	4.41E-02	3.01E-02	2.88E-02
	Std	1.04E-03	2.60E-01	8.78E-03	9.04E-04	7.39E-04	4.12E-03	5.27E-04	6.99E-04
	Best	1.86E-02	5.50E-02	2.91E-13	2.27E-02	2.12E-02	3.53E-02	2.92E-02	2.76E-02
Diabetes	Avg	1.40E-01	2.54E-01	3.99E-02	1.48E-01	1.43E-01	1.87E-01	1.53E-01	1.53E-01
	Std	2.63E-03	8.78E-02	4.09E-02	1.62E-03	1.25E-03	1.00E-02	1.20E-03	8.19E-04
	Best	1.36E-01	2.15E-01	1.38E-02	1.46E-01	1.40E-01	1.73E-01	1.52E-01	1.52E-01
Habitit	Avg	3.82E-02	2.39E-01	6.18E-02	4.87E-02	4.11E-02	1.21E-01	8.34E-02	7.27E-02
	Std	6.84E-03	1.32E-01	6.57E-02	5.39E-03	2.86E-03	8.72E-03	2.80E-03	3.61E-03
	Best	2.62E-02	1.12E-01	1.98E-12	3.90E-02	3.56E-02	1.06E-01	7.84E-02	6.61E-02
Vertebral	Avg	8.66E-02	2.96E-01	8.12E-02	8.95E-02	9.00E-02	1.48E-01	1.07E-01	9.62E-02
	Std	1.42E-03	1.61E-01	1.37E-01	9.65E-04	1.38E-03	1.42E-02	4.75E-03	1.81E-03
	Best	8.36E-02	1.42E-01	4.90E-03	8.81E-02	8.80E-02	1.23E-01	1.01E-01	9.28E-02
Liver	Avg	1.76E-01	3.55E-01	5.72E-02	1.88E-01	1.84E-01	2.41E-01	2.15E-01	2.02E-01
	Std	3.06E-03	1.28E-01	6.50E-02	2.87E-03	3.48E-03	6.86E-03	1.70E-03	1.58E-03
	Best	1.73E-01	2.29E-01	1.76E-02	1.82E-01	1.79E-01	2.30E-01	2.13E-01	1.99E-01
Diagnosis I	Avg	3.36E-12	1.59E-01	1.90E-12	9.24E-11	2.79E-09	2.37E-02	5.05E-05	0.00
	Std	9.17E-12	1.49E-01	1.08E-12	1.17E-10	4.23E-09	2.27E-02	6.66E-05	0.00
	Best	1.61E-14	2.21E-02	6.72E-13	3.42E-12	4.75E-12	5.06E-04	8.81E-08	0.00
Diagnosis II	Avg	2.78E-13	1.10E-01	8.99E-02	2.80E-13	1.28E-10	1.15E-02	3.54E-07	0.00
	Std	7.93E-13	1.24E-01	2.26E-01	4.51E-13	3.85E-10	1.19E-02	4.43E-07	0.00
	Best	1.69E-16	1.22E-02	9.34E-13	8.55E-15	1.88E-13	2.17E-04	1.20E-08	0.00
Parkinson	Avg	3.54E-02	1.88E-01	7.66E-02	5.81E-02	4.05E-02	1.28E-01	8.75E-02	6.78E-02
	Std	1.35E-02	6.91E-02	1.20E-01	6.92E-03	4.89E-03	1.31E-02	3.21E-03	5.96E-03
	Best	2.09E-02	1.09E-01	2.99E-13	4.68E-02	3.39E-02	1.05E-01	8.31E-02	5.90E-02

features are Mean Corpuscular Volume (MCV), alkaline phosphotase (ALKPHOS), alamine aminotransferase (SGPT), aspartate aminotransferase (SGOT) and gamma-glutamyl transpeptidase (GAMMAGT). The sixth feature is the number of alcoholic beverage drinks per day (DRINKS).

– Acute Inflammations Dataset (Diagnosis-I and Diagnosis-II): The data was constructed by a medical expert for the task of presumptive diagnosis of two diseases of the urinary system (acute inflammation of urinary bladder (Diagnosis-I) and acute Nephritis of renal pelvis origin (Diagnosis-II)). Each of the 120 instances in the dataset represents a potential patient. There are six features in the dataset for each potential patient. One feature is the the temperature of patient which is a real number while the other five features are binary which are Occurrence of nausea, Lumbar pain, Urine pushing, Micturition pains and Burning of urethra, itch, swelling of urethra outlet [4].
– Parkinson: This dataset was collected by Max Little of the University of Oxford, in collaboration with the National Centre for Voice and Speech, Denver, Colorado. The task of the data is to classify healthy people from those with Parkinson disease. The dataset has 23 of biomedical voice measurements from 31 people, 23 with Parkinsons disease. Total number of voice recordings is 195. There are almost 6 recordings for each person [13].

5.3 Results

To evaluate the proposed MVO technique, we compare the MVO results with standard BP, LM and other metaheuristic techniques by quantifying the accuracy and MSE evalua-

tion measures. Table 3 shows the average accuracy with standard deviation for 10 runs, as well as the best accuracy result of the proposed MVO, BP, LM, GA, PSO, DE, FF and CS on the given datasets. As per the results, MVO outperforms all other optimizers in Blood, Breast cancer, Diabetes, Hepatitis, Liver, and Parkinson datasets with an average accuracy of 0.796, 0.973, 0.768, 0.894, 0.725, and 0.930, respectively. Moreover, MVO has an average accuracy of 1.00 for Diagnosis-I and Diagnosis-II, which are the same results for GA, PSO, Firefly and CS. In addition, MVO shows improvements compared to the other optimizers considering the best accuracy results brackets. It can also be noticed that MVO has small standard deviation for all datasets, which indicates that MVO is robust and stable.

Table 4 shows the average of MSE, standard deviation, and best values of MSE results for all datasets. Inspecting the table of results, it can be seen that the MVO outperforms other techniques in two datasets: Habiti and Parkinson. Moreover, it is ranked second for the rest of datasets and shows very competitive results compared to the LM algorithm. The standard deviation values indicate that MVO provides very small values which proves the efficiency and robustness of this algorithm. In summary, MVO shows very competitive optimization results of the set of weights and biases.

Convergence curves for all metaheuristic optimizers are shown in the Fig. 5. The convergence curves show the averages of 10 independent runs over the course of 200 iterations. The figure shows that MVO has the fastest convergence speed on the Liver and Parkinson dataset. For other datasets, MVO provides very close performance compared to the GA technique. Moreover, MVO has the lowest values of the MSE for most of datasets compared to the other metaheuristic algorithms. These results show that MVO has better optimization efficiency and faster convergence performance than GA, PSO, DE, Firefly and CS.

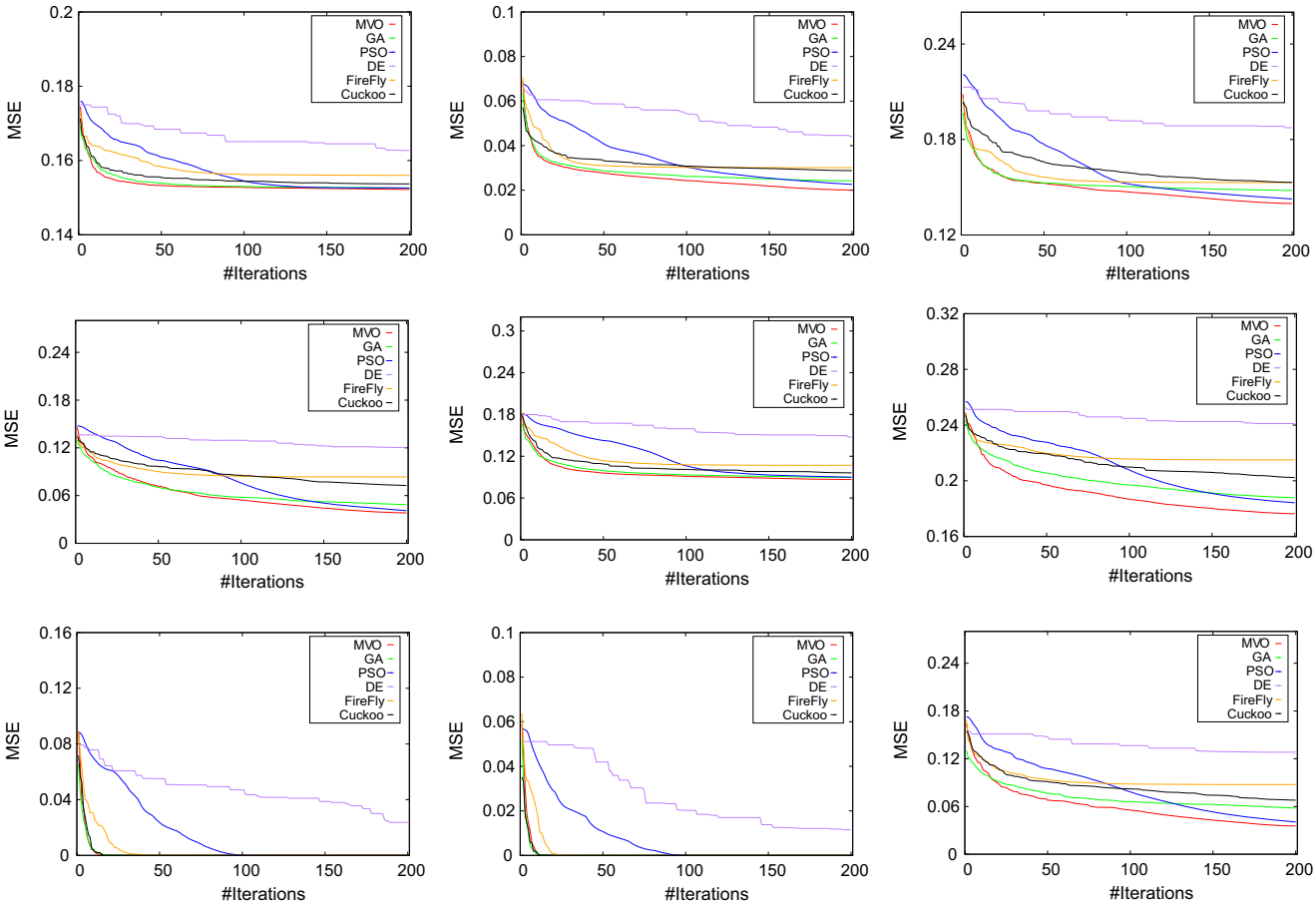

Fig. 5 MSE convergence curves of different classification datasets. Figure 5a–i MSE Convergence curve for Blood, Breast cancer, Diabetes, Hepatitis, Vertebral, Liver, Diagnosis-I, Diagnosis-II, and Parkinson datasets, respectively

Figure 6 shows the Boxplot relative to 10 runs of MVO, BP, LM, GA, PSO, DE, Firefly and CS. The Boxplots are used to analyze the the MLP optimizer variability in getting MSE values in all the runs. As shown in the figure, the Boxplots confirm and justify the better performance of MVO when training MLPs.

The above-discussed results showed that MVO outperforms other algorithms in average. To evaluate the overall performance of MVO against other optimization algorithms in each independent run and confirm the significance of the results, we conducted Friedman statistical test. This statistical is done by ranking the different techniques (MVO, BP, LM, GA, PSO, DE, Firefly and CS) based on the average accuracy values for each dataset. Table 5 shows the average ranks obtained by each optimization technique in the Friedman test. Table 5 shows that significant differences exist between the 8 techniques (the lower is better). MVO has highest overall ranking in comparison with other techniques.

This comprehensive comparative study showed that the MVO algorithm has merits among the current trainers in the literature. Problem of training MLP is very challenging and has a large number of local solutions. The better performance of the MVO algorithm is due to the high local optima avoidance which originates from the structure of this algorithm. The white-black hole tunnel causes abrupt changes in the weights/biases of MLP and results in boosting exploration of the search space and local optima avoidance. In addition, the search space of training problem changes for every dataset. MVO performed very well in all of them, which shows how flexible this algorithm is for solving different problems with diverse search spaces. This is due to the gradient-free mechanism of this trainer which considers the training problem as a black box.

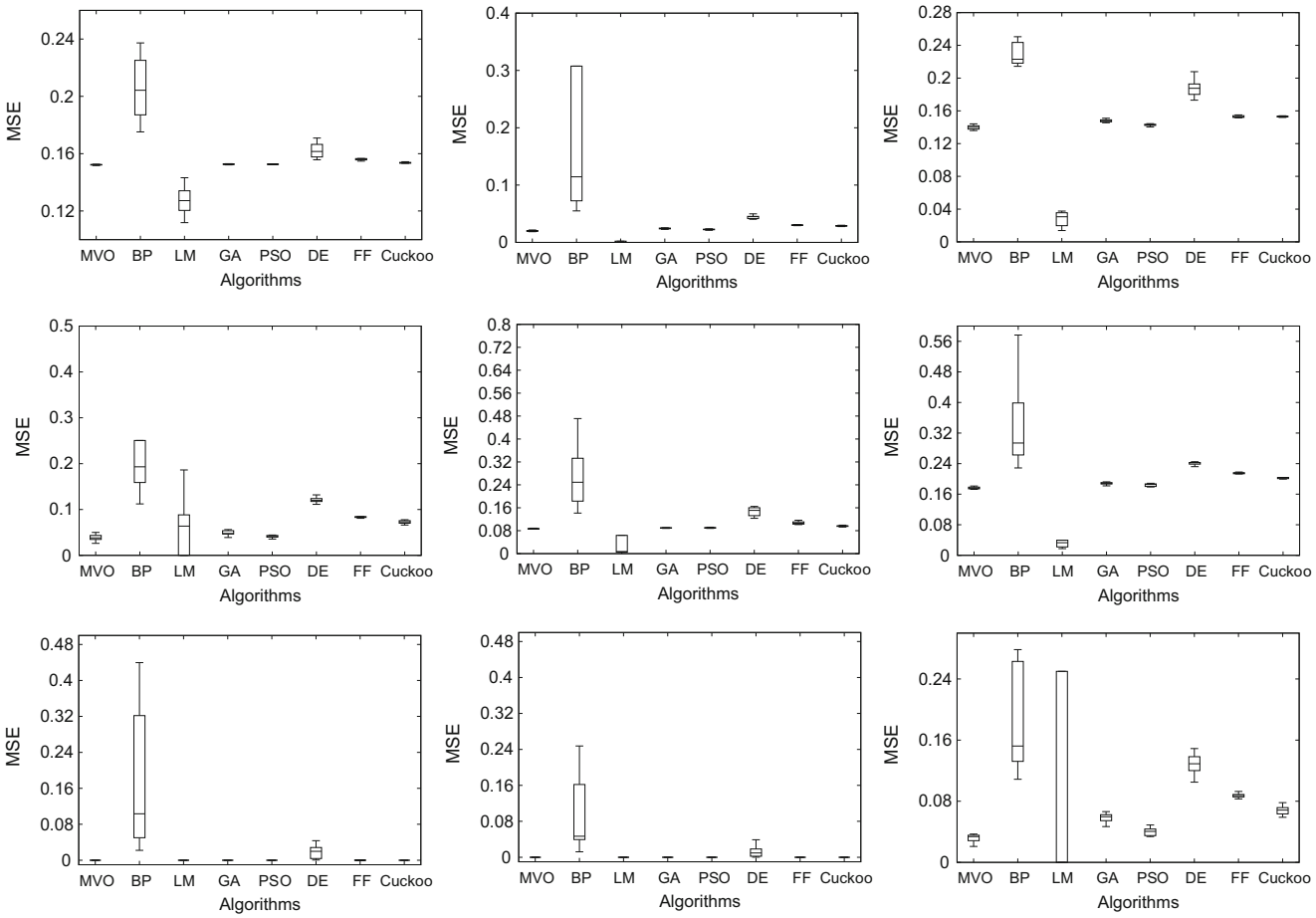

Fig. 6 Boxplot representation of the MSE for MVO, BP, LM, GA, PSO, DE, Firefly and CS on different datasets. Figure 6a–i MSE Boxplot for Blood, Breast cancer, Diabetes, Hepatitis, Vertebral, Liver, Diagnosis-I, Diagnosis-II, and Parkinson datasets, respectively

Table 5 Average Rankings of the techniques (Friedman)

Algorithm	Ranking
MVO	1.8889
BP	8.0
LM	5.7778
GA	2.7222
PSO	2.7778
DE	6.6667
Firefly	3.8889
CS	4.2778

6 Conclusion

In this paper, we have proposed a new training approach based on MVO to train MLP neural network. The training approach took into account the capabilities of the MVO in terms of high exploration and exploitation to locate the optimal values for weights and biases of MLPs. Experimental results on nine classification problems with different characteristics show that our proposed approach is efficient to train MLPs compared to well-known training methods such as BP, LM, GA, PSO, DE, Firefly and CS that have been used in the literature. The statistical results of MSE over 10 runs show that our proposed approach is robust since the variances are relatively very small. Furthermore, the accuracy of the results obtained for weights and biases is very high and outperform other techniques. In addition, the significance of the results was statistically confirmed using Friedman test and compared with BP, LM, GA, PSO, DE, Firefly and CS. As an overall outcome, MVO has the highest rank among all the techniques employed.

Our future research will include the investigation of our proposed approach on other types of problems such as multi-class classification. We will also investigate the effectiveness of our proposed algorithm with larger datasets. In addition, we are planing to check the performance of MVO in training other types of neural networks such as Radial Basis Function (RBF) neural network.

References

1. Basheer I, Hajmeer M (2000) Artificial neural networks: fundamentals, computing, design, and application. J Microbiol Methods 43(1):3–31
2. Braik M, Sheta A, Arieqat A (2008) A comparison between gas and pso in training ann to model the te chemical process reactor. In: AISB 2008 Convention communication, interaction and social intelligence, vol 1. citeseer, p 24
3. Chaves AdCF, Vellasco MMB, Tanscheit R (2005) Fuzzy rule extraction from support vector machines. In: Hybrid Intelligent Systems, 2005. HIS'05. Fifth International Conference on. IEEE, pp 6–pp
4. Czerniak J, Zarzycki H (2003) Application of rough sets in the presumptive diagnosis of urinary system diseases. In: Artificial Intelligence and Security in Computing Systems. Springer, pp 41–51
5. Fausett LV (1994) Fundamentals of Neural Networks: Architectures, Algorithms, and Applications. Prentice Hall
6. Gupta JN, Sexton RS (1999) Comparing backpropagation with a genetic algorithm for neural network training. Omega 27(6):679–684
7. Holland JH (1992) Adaptation in natural and artificial systems. MIT press cambridge, MA, USA
8. Hornik KJ, Stinchcombe D, White H (1989) Multilayer feedforward networks are universal approximators. Neural Netw 2(5):359–366
9. Khan K, Sahai A (2012) A comparison of ba, ga, pso, bp and lm for training feed forward neural networks in e-learning context. Int J Intell Syst Appl (IJISA) 4(7):23
10. Kowalski P, Ukasik S (2015) Training neural networks with krill herd algorithm. Neural Processing Letters pp 1–13, doi:10.1007/s11063-015-9463-0
11. Lari N, Abadeh M (2014) Training artificial neural network by krill-herd algorithm. In: IEEE 7th Joint Information Technology and Artificial Intelligence Conference (ITAIC), 2014 International, pp 63–67. doi:10.1109/ITAIC.2014.7065006
12. Lichman M (2013) UCI machine learning repository. http://archive.ics.uci.edu/ml
13. Little MA, McSharry PE, Roberts SJ, Costello DA, Moroz IM et al. (2007) Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection. BioMed Eng OnLine 6(1):23
14. Liu Z, Liu A, Wang C, Niu Z (2004) Evolving neural network using real coded genetic algorithm (ga) for multispectral image classification. Futur Gener Comput Syst 20(7):1119–1129
15. Mangasarian OL, Setiono R, Wolberg W (1990) Pattern recognition via linear programming: Theory and application to medical diagnosis. Large-scale numerical optimization, pp 22–31
16. Mendes R, Cortez P, Rocha M, Neves J (2002) Particle swarms for feedforward neural network training. In: Proceedings of the 2002 International Joint Conference on Neural Networks, 2002. IJCNN '02, vol 2, pp 1895–1899. doi:10.1109/IJCNN.2002.1007808
17. Mirjalili S (2015) How effective is the grey wolf optimizer in training multi-layer perceptrons. Appl Intell 43(1):150–161. doi:10.1007/s10489-014-0645-7
18. Mirjalili S, Hashim SZM, Sardroudi HM (2012) Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm. Appl Math Comput 218(22):11,125–11,137
19. Mirjalili S, Mirjalili SM, Lewis A (2014) Let a biogeography-based optimizer train your multi-layer perceptron. Inf Sci 269:00747. doi:10.1016/j.ins.2014.01.038. http://www.sciencedirect.com/science/article/pii/S00200255140
20. Mirjalili S, Mirjalili S, Hatamlou A (2015) Multi-verse optimizer: a nature-inspired algorithm for global optimization. Neural Computing and Applications:1–19. doi:10.1007/s00521-015-1870-7
21. Montana DJ, Davis L (1989) Training feedforward neural networks using genetic algorithms. In: Proceedings of the 11th International Joint Conference on Artificial Intelligence - Volume 1, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, IJCAI'89, pp 762–767, http://dl.acm.org/citation.cfm?id=1623755.1623876
22. Nayak J, Naik B, Behera H (2015) A novel nature inspired firefly algorithm with higher order neural network: Performance analysis. Engineering Science and Technology, an International Journal doi:10.1016/j.jestch.2015.07.005
23. Rao M (2000) Feedforward neural network methodology. Technometrics 42(4):432–433

24. Rumelhart DE, Hinton GE, Williams RJ (1988) Neurocomputing: Foundations of research. MIT Press, Cambridge, MA, USA, chap Learning Representations by Back-propagating Errors, pp 696–699, http://dl.acm.org/citation.cfm?id=65669.104451
25. Seiffert U (2001) Multiple layer perceptron training using genetic algorithms. In: Proceedings of the European Symposium on Artificial Neural Networks, Bruges, Bélgica
26. Sexton RS, Gupta JN (2000) Comparative evaluation of genetic algorithm and backpropagation for training neural networks. Inf Sci 129(1–4):00682. doi:10.1016/S0020-0255(00)00068-2. http://www.sciencedirect.com/science/article/pii/S00200255000
27. Sexton RS, Dorsey RE, Johnson JD (1998) Toward global optimization of neural networks: a comparison of the genetic algorithm and backpropagation. Decis Support Syst 22(2):00407. doi:10.1016/S0167-9236(97)00040-7. http://www.sciencedirect.com/science/article/pii/S01679236970
28. Slowik A, Bialko M (2008) Training of artificial neural networks using differential evolution algorithm. In: Conference on Human System Interactions, 2008, IEEE, pp 60–65
29. Valian E, Mohanna S, Tavakoli S (2011) Improved cuckoo search algorithm for feedforward neural network training. Int J Artif Intell Appl 2(3):36–43
30. Wdaa ASI (2008) Differential evolution for neural networks learning enhancement. PhD thesis, Universiti Teknologi Malaysia
31. Wolberg WH, Mangasarian OL (1990) Multisurface method of pattern separation for medical diagnosis applied to breast cytology. Proceed Nat Acad Sci 87(23):9193–9196
32. Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. IEEE Trans Evol Comput 1(1):67–82
33. Wu WL, Su FC, Cheng YM, Chou YL (2001) Potential of the genetic algorithm neural network in the assessment of gait patterns in ankle arthrodesis. Ann Biomed Eng 29(1):83–91
34. Yeh IC, Yang KJ, Ting TM (2009) Knowledge discovery on rfm model using bernoulli sequence. Expert Syst Appl 36(3):5866–5871
35. Yu J, Wang S, Xi L (2008) Evolving artificial neural networks using an improved PSO and DPSO. Neurocomputing 71(4–6):1054 – 1060. doi:10.1016/j.neucom.2007.10.013. http://www.sciencedirect.com/science/article/pii/S0925231207003591, neural Networks: Algorithms and Applications50 Years of Artificial Intelligence: a Neuronal Approach4th International Symposium on Neural NetworksCampus Multidisciplinary in Perception and Intelligence

Hossam Faris is an Associate professor at Business Information Technology department, King Abdullah II School for Information Technology, The University of Jordan (Jordan). Currently he is also a Postdoctoral researcher at the Information and Communication Technologies Research Center (CITIC), University of Granada (Spain). Hossam Faris received his BA, M.Sc. degrees (with excellent rates) in Computer Science from Yarmouk University and Al-Balqa‘ Applied University in 2004 and 2008 respectively in Jordan. Since then, he has been awarded a full-time competition-based PhD scholarship from the Italian Ministry of Education and Research to peruse his PhD degrees in e-Business at University of Salento, Italy, where he obtained his PhD degree in 2011. His research interests include: Applied Computational Intelligence, Evolutionary Computation, Knowledge Systems, Semantic Web and Ontologies.

Ibrahim Aljarah is an Assistant Professor of Computer Science at the University of Jordan - Department of Business Information Technology, Jordan. He obtained his bachelor degree in Computer Science from Yarmouk University - Jordan, 2003. Ibrahim also obtained his master degree in Computer Science and Information Systems from the Jordan University of Science and Technology - Jordan in 2006. He also obtained his Ph.D. in Computer Science from the North Dakota State University (NDSU), USA, in May 2014. He has published more than 20 papers in refereed international conferences and journals. His research focuses on Data mining, Machine Learning, Big Data, MapReduce, Hadoop, Swarm intelligence, Evolutionary Computation, Social Network Analysis (SNA), and large scale distributed algorithms. In addition, his research aims to make use of the nature-inspired approaches in data mining applications to be efficient and powerful for big data. Furthermore, His research benefits from the MapReduce methodology as a big data processing model to build scalable data mining algorithms.

Seyedali Mirjalili is a lecturer in Griffith College, Griffith University. He received his B.Sc. degree in Computer Engineering (software) from Yazd University, M.Sc. degree in Computer Science from Universiti Teknologi Malaysia (UTM), and Ph. D. in Computer Science from Griffith University. He was a member of Soft Computing Research Group (SCRG) at UTM. His research interests include Robust Optimisation, Multi-objective Optimisation, Swarm Intelligence, Evolutionary Algorithms, and Artificial Neural Networks. He is working on the application of multi-objective and robust meta-heuristic optimisation techniques in Computational Fluid Dynamic (CFD) problems as well.