# A hybrid Evolutionary Functional Link Artificial Neural Network for Data mining and Classification

Faissal MILI
Applied Economics and Simulation,
Faculty of Management and Economic Sciences,
5100 Mahdia, Tunisia

Manel HAMDI
International Finance Group Tunisia , Faculty of
Management and Economic Sciences of Tunis, Tunisia
2092, El Manar Tunisia.

***Abstract*— This paper presents a specific structure of neural network as the functional link artificial neural network (FLANN). This technique has been employed for classification tasks of data mining. In fact, there are a few studies that used this tool for solving classification problems. In this present research, we propose a hybrid FLANN (HFLANN) model, where the optimization process is performed using 3 known population based techniques such as genetic algorithms, particle swarm and differential evolution. This model will be empirically compared to FLANN based back-propagation algorithm and to others classifiers as decision tree, multilayer perceptron based back-propagation algorithm, radical basic function, support vector machine, and K-nearest Neighbor. Our results proved that the proposed model outperforms the other single model.**

***Keywords- component Data mining; Classification; Functional link artificial neural network; genetic algorithms; Particle swarm; Differential evolution.***

## I. INTRODUCTION

Classification task is a very important topic in data mining. A lot of research ([1], [2], [3]) has focused on the field over the last two decades. The Data mining is a knowledge discovery process from large databases. The extracted knowledge will be used by a human user for supporting a decision that is the ultimate goal of data mining. Therefore, classification decision is our aim in this study. A various classification models have been used in this regard. M. James [4] has employed a linear/quadratic discriminates techniques for solving classification problems. Another procedure has been applied using decision trees ([5], [6]). In the same context, Duda et al. [7] have proposed a discriminant analysis based on the Bayesian decision theory. Nevertheless, these traditional statistical models are built mainly on various linear assumptions that will be necessary satisfied. Otherwise, we cannot apply these techniques for classification tasks. To overcome the disadvantage, artificial intelligent tools have been emerged to solve data mining classification problems. For this purpose, genetic algorithms models were used [8]. In a recent research, Zhang ([9], [10]) have introduced the neural networks technique as a powerful classification tool. In these studies, he showed that neural network is a promising alternative tool compared to various conventional classification techniques. In a more recent literature, a specific structure of neural network has been employed for classification task of data mining as the functional link artificial neural network (FLANN). In fact, there are a few studies ([11], [12], [13]) used this tool for solving classification problems.

In this present research, we propose a hybrid FLANN (HFLANN) model based on three metaheuristics population based optimization tools such: genetic algorithms (GAs), particle swarm optimization (PSO) and differential evolution. This model will be compared to the trained FLANN based backpropagation and multilayer perceptron (MLP) as the most famous model in the area.

## II. CONCEPTS AND DEFINITION

### *A. Population based algorithms*

Population based algorithms are classed as a computational intelligence techniques representing a class of robust optimization ones. These population based ones make use of a population of solution in the same time based on natural evolution.

Many population based algorithms are presented in the literature such evolutionary programming [14], evolution strategy [15], genetic algorithms [16], genetic programming [17], Ant Colony [18], particle swarm [19] and differential evolution [20]. These algorithms differ in selection, offspring generation and replacement mechanisms. Genetic algorithms, particle swarm and differential evolutions represent the most popular ones.

#### *1) Genetic algorithms*

Genetic algorithms (GAs) are defined as a search technique that was inspired from Darwinian Theory. The idea is based on the theory of natural selection. We assume that there is a population composed with different characteristics. The stronger will be able to survive and they pass their characteristics to their offsprings.

The total process is described as follows:

1- Generate randomly an initial population;

2- Evaluate this population using the fitness function;

3- Apply genetic operators such selection, crossover, and mutation;

4- Turn the process “Evaluation Crossover mutation” until reaching the stopped criteria fixed in prior.

### 2) *Particle swarm*

Presented in 1995 by L. Kennedy and R. Eberhart [19], particle swarm optimization (PSO) represents one of the most known population-based approaches, where particles change their positions with time. These particles fly around in a multidimensional search space, and each particle adjusts its position according to its own experience and the experience of their neighboring, making use of the best position encountered by itself and its neighbors. The direction of a particle is defined by the set of neighboring and its correspondent history of experience.

An individual particle i is composed of three vectors:

- Its position in the V-dimensional search space

$\vec{X_i} = (X_{i1}, X_{i2}, \dots, X_{iV})$

- The best position that it has individually found $\vec{P_i} = (P_{i1}, P_{i2}, \dots, P_{iV})$

- Its velocity $\vec{V_i} = (V_{i1}, V_{i2}, \dots, V_{iV})$

Particles were originally initialized in a uniform random manner throughout the search space; velocity is also randomly initialized.

These particles then move throughout the search space by a fairly simple set of update equations. The algorithm updates the entire swarm at each time step by updating the velocity and position of each particle in every dimension by the following rules:

$$. V_{iD} = \chi * (W * V_{iN} + C * \varepsilon_1(P_{nV} - X_{iV}) + C * \varepsilon_2(P_{gV} - X_{iV})) \quad (1)$$

$$X_{iN} = X_{iV} + V_{iV} \quad (2)$$

Where in the original equations:

C is a constant with the value of 2.0

$\varepsilon_1$ and $\varepsilon_2$ are independent random numbers uniquely generated at every update for each individual dimension (n = 1 to V).

$P_{gV}$ is the best position found by the global population of particle.

$P_{nV}$ is the best position found by any neighbor of the particle.

W: the weight

$\chi$ : the constriction factor.

### 3) *Differential evolution*

Proposed by Storn and Price in 1995 [20], differential evolution represents a new floating evolutionary algorithm using a special kind of differential operator. Easy implementation and negligible parameter tuning makes this algorithm quite popular.

Like any evolutionary algorithm, differential evolution starts with a population. Differential evolution is a small and simple mathematical model of a big and naturally complex process of evolution. So, it is easy and efficient.

Firstly, there are five DE strategies (or schemes) that were proposed by R. Storn and K. Price [20]:

- **Scheme DE/rand/1 :**

$$\omega = x_1 + F^* (x_2 - x_3) \quad (3)$$

- **Scheme DE/rand/2 :**

$$\omega = x_5 + F * (x_1 + x_2 - x_3 - x_4) \quad (4)$$

- **Scheme DE/best/1:**

$$\omega = x_{best} + F * (x_1 - x_2) \quad (5)$$

- **Scheme DE/best/2:**

$$\omega = x_{best} + F * (x_1 + x_2 - x_3 - x_4) \quad (6)$$

- **Scheme DE/rand-to best/1:**

$$\omega = x+\lambda^* (x_{best}-x_1)+F * (x_2-x_3) \quad (7)$$

Later, two more strategies were introduced [21].

We present the trigonometric scheme defined by:

$$\omega = (x_1 + x_2 + x_3)/3 + (p_2 - p_1) * (x_1 - x_2) + (p_3 - p_2) * (x_2 - x_3) + (p_1 - p_3) * (x_3 - x_1) \quad (8)$$

$$p_i=|f(x_i)/ (f(x_1) + f(x_2) + f(x_3)) |, i= 1, 2, 3 ; \quad (9)$$

F define the constriction factor generally taken equal to 0.5

x define the selected element

$x_1$, $x_2$, $x_3$, $x_4$ and $x_5$ represent random generated elements from the population.

Many others schemes can be found in the literature [20].

## B. *Functional Link Artificial Neural Networks*

The FLANN architecture was originally proposed by Pao et al. [22]. The basic idea of this model is to apply an expansion function which increases the input vector dimensionality. We say that the hyper-planes generated provide greater discrimination capability in the input pattern space. By applying this expansion, we needn't the use of the hidden layer, making the learning algorithm simpler. Thus, compared to the MLP structure, this model has the advantage to have faster convergence rate and lesser computational cost.

The conventional nonlinear functional expansions which can be employed are trigonometric, power series or Chebyshev type. R. Majhi et al. [23], shows that use of trigonometric expansion provides better prediction capability of the model. Hence, in the present case, trigonometric expansion is employed.

Let each element of the input pattern before expansion be represented as X(i), 1 < i < I where each element x(i) is functionally expanded as Zn(i) , 1 < n < N , where N = number of expanded points for each input element. In this study, we take N=5.

I= the total number of features

As presented in figure 1, the expansion of each input pattern is done as follows.

$$Z_1(i) = X(i),\ Z_2(i) = f_1(X(i)),....,\ Z_5(i) = f_5(X(i)) \qquad (10)$$

These expanded inputs are then fed to the single layer neural network and the network is trained to obtain the desired output.

## III. Hybrid FLANN description

The proposed hybrid FLANN is based on evolutionary algorithms as genetic algorithms, particle swarm and differential evolution.

### A. *Resampling technique:*

In order to avoid overfitting, we use the (2*5) K fold cross-validation resampling technique. We proceed as follows:

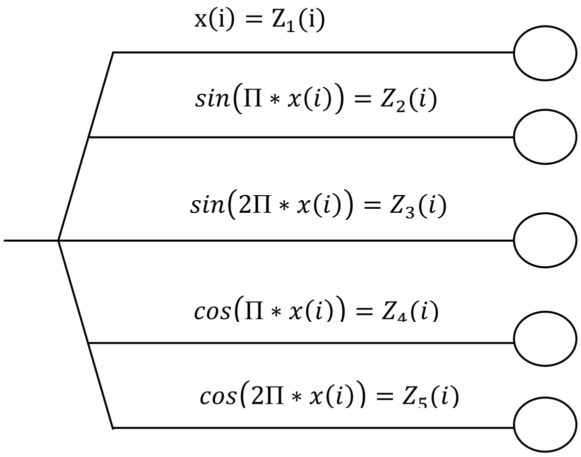

$x(i) = Z_1(i)$
$sin(\Pi * x(i)) = Z_2(i)$
$sin(2\Pi * x(i)) = Z_3(i)$
$cos(\Pi * x(i)) = Z_4(i)$
$cos(2\Pi * x(i)) = Z_5(i)$

Figure 1. Functional expansion of the first element

We divide initial database into 5 folds (K=5) where each one contain the same repartition of classes. For example, if initial population contains 60% of class 1 and 40% of class 2, then all the resulted K folds must have the same repartition.

### B. *Generation*

We begin the process by generating randomly initial solution. We execute partial training using differential evolution in order to improve initial state.

### C. *Fitness fuction and evaluation*

In order to evaluate each solution, two criterions are used such the mean square error (MSE) and the misclassification error (MCE) rate. If we have to compare solutions A and B, we apply the following rules: A is preferred to B If and only if MCE(A)< MCE(B) Or MCE(A)= MCE(B) and MSE(A)< MSE(B).

### D. *Selection*

Many selections are defined in the literature such the Roulette wheel method, the N/2 elitist method and the tournament selection method. The last method will be used here. The principle is to compare two solutions, and the best one will be selected.

N/2 elitist is used at the beginning of the process in order to select 50% of generated solution.

### E. *Crossover*

Two parents are selected randomly in order to exchange their information. Two crossovers are applied and described as follows:

#### 1) *Crossover 1 (over input feature):*

An input feature is chosen randomly to exchange his correspondent weight between the selected two parents.

#### 2) *Crossover 2 (over output nodes):*

An output is chosen randomly to exchange his correspondent weight.

#### 3) *Crossover 3 (Crossover over connection):*

A connection position is chosen randomly and his correspondent weight is exchanged between the two parents.

### F. *Mutation*

#### 1) *Mutation 1(over connection)*

A connection position is chosen randomly and his correspondent weight has been controlled. If this connection is connected, his correspondent weight is disconnected by setting his value equal to zero. Else, this connection is connected.

#### 2) *Mutation 2 (over one input feature)*

An input feature is chosen randomly and his correspondent weights have been controlled. If this input feature is connected (there is at least one weights of his correspondent ones is different from zero), it will be disconnected by putting all his entire weight equal to zero. Else if this input feature is totally disconnected, it will be connected there by generating weights different from zero.

#### 3) *Mutation 3 (over two input feature)*

We do the same like mutation 2 but here simultaneously for the two selected features.

#### 4) *Mutation 4 ( over three input feature)*

In this mutation, the same principle is used for three input features.

We note that many input features connection and disconnection can be executed in the same time when having a large number of features. This crossover helps to remove undesirable features from our classification process and can improve the final performance process.

### G. *Particle swarm optimization (PSO)*

In the presented paper, we define three PSO model based on the notion of neighbor.

#### 1) *PSO based on resulted genetic offsprings*

First, we apply genetic operators. Each offspring that improve our fitness function define a neighbor, and used in equation (1).

#### 2) *PSO based on Euclidian distance:*

For each particle, we compute the Euclidian distance between this particle and the rest of the population. Next we choose the five nearest particles based on this distance.

From the selected subset of neighbors, we choose the best one which has the best fitness value. This selected one defines our neighbor to be replaced in equation (1).

#### 3) *PSO based on the last best visited solution:*

In this case, each particle flies and memorizes his best reached solution. This memory defines the neighbor to be used in equation (1).

### *H. Differential evolution*

In this work, we proceed as follows:

- First, for each candidate x, we generate five random solution $x_1$, $x_2$, $x_3$, $x_4$ and $x_5$.

- Next we apply seven chosen schemes as follows:

DE1: Scheme DE/direct/1 :

$$\omega = x + F^* (x_2 - x_1) \qquad (11)$$

DE2: Scheme DE/best/1 :

$$\omega = x_{best} + F^* (x_2 - x_1) \qquad (12)$$

DE3: Scheme DE/best/1 :

$$\omega = x_{best} + F^* (x_3 - x_2) \qquad (13)$$

DE4: Scheme DE/best/1 :

$$\omega = x_{best} + F^* (x_3 - x_1) \qquad (14)$$

DE5: Scheme DE/best/2 :

$$\omega = x_{best} + F * (x1 + x2 - x3 - x4) \qquad (15)$$

DE6: Scheme DE/rand/2 :

$$\omega = x5 + F * (x1 + x2 - x3 - x4) \qquad (16)$$

DE7: with Trigonometric Mutation:

$$\omega = (x1 + x2 + x3)/3 + (p2 - p1) * (x1 - x2) + (p3 - p2) * (x2 - x3) + (p1 - p3) * (x3 - x1) \qquad (17)$$

$$p_i = |f(x_i)/ (f(x1) + f(x2) + f(x3)) |, i= 1, 2, 3 ; \qquad (18)$$

### *I. Stopping criterion:*

The process turns in a cycle until reaching a maximum number of epochs without any improvement. We fix the maximum number of epochs equal to 30 epochs.

## IV. EXPERIMENTAL STUDIES:

11 real-world databases were selected there to be used in simulation works. They are chosen from the UCI repository machine learning, which is commonly used to benchmark learning algorithms [24].

We compare the results of the proposed hybrid FLANN (HFLANN) with FLANN based on the gradient descent algorithm. Next, Comparison with other classifiers will be done.

### *A. Description of the databases*

A brief description of used databases for experimental setup is presented in table I. Num. is the numeric features, Bin. is the binary ones, and Nom. is the nominal inputs that mean discrete with three or more distinct labels.

TABLE I. SUMMARY OF THE DATASET USED IN SIMULATION STUDIES

| **Dataset** | **Inputs** | | | | **Ex.** | **Cls** |
|---|---|---|---|---|---|---|
| | **Num.** | **Bin.** | **Nom.** | **Total** | | |
| **IRIS** | 4 | 0 | 0 | 4 | 150 | 3 |
| **VOTING** | 0 | 16 | 0 | 16 | 435 | 2 |
| **BREAST** | 0 | 0 | 9 | 9 | 699 | 2 |
| **PRIMA** | 8 | 0 | 0 | 8 | 768 | 2 |
| **CREDIT** | 6 | 4 | 4 | 14 | 690 | 2 |
| **BALANCE** | 4 | 0 | 0 | 4 | 625 | 3 |
| **WINE** | 13 | 0 | 0 | 13 | 178 | 3 |
| **BUPA** | 6 | 0 | 0 | 6 | 345 | 2 |
| **ECOLI** | 7 | 0 | 0 | 7 | 336 | 8 |
| **GLASS** | 10 | 0 | 0 | 10 | 214 | 6 |
| **ZOO** | 1 | 15 | 0 | 16 | 101 | 7 |

### *B. Convergence test:*

In order to test the convergence of the proposed hybrid FLANN, a comparison will be done with trained FLANN using the back-propagation algorithm. Results are presented in figure 2 and figure 3. Comparison is done based on the required time and number of epochs for convergence.

From figure 2, we find that our process needs less than 200 seconds 20 epochs to converge. Figure 3 present results for FLANN based on back-propagation. This model requires less than 150 seconds and 15 epochs to converge.

The proposed hybrid FLANN has a strong ability to converge fast and requires approximately the same time and epochs than FLANN based back-propagation.

### *C. Comparative results:*

The classification accuracy of the proposed hybrid FLANN are compared with the results obtained from FLANN trained using the back-propagation algorithm. Results are presented in table II. Bold typeface is used to highlight the results that are significantly better.

We find that the proposed model gives better results for all used databases. So, our proposed evolutionary process trains better than the back-propagation algorithm.

Next, a basic comparison between the HFLANN and five others classifiers is performed using fourth databases. These classifiers are:

- The decision tree based C.45,

- The multilayer perceptron (MLP) based back-propagation algorithms,

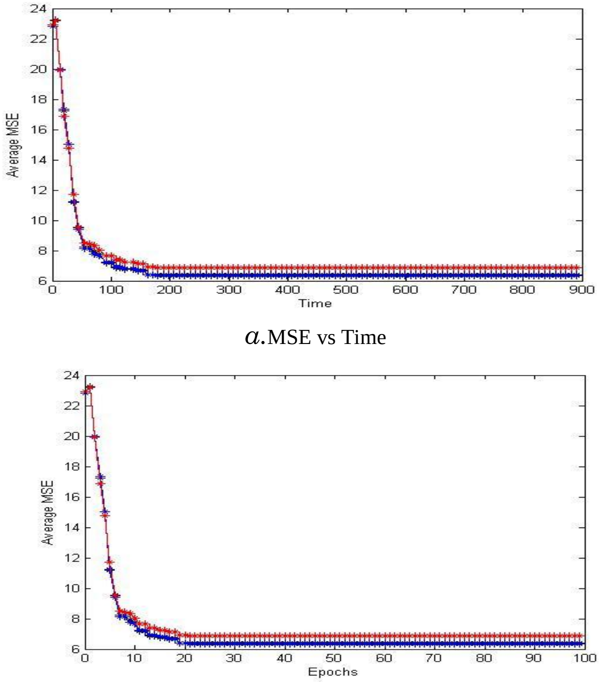
- The radical basic function (RBF),

- The support vector machine (SVM),

- The K-nearest Neighbor (KNN),

Results are presented in table II. From this table, we find that HFLANN is better for 3 databases and SVM is better for the forth. We say that HFLANN is a good classifier giving better results in the majority of used databases.

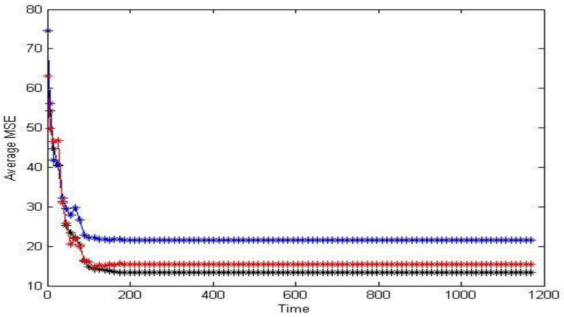
Table IV presents the number of different local and global improvement of used population based algorithms. We find that PSO represents the best local population based technique with 104476 improvements, and differential evolution is the best global one with 1854 of improvement.



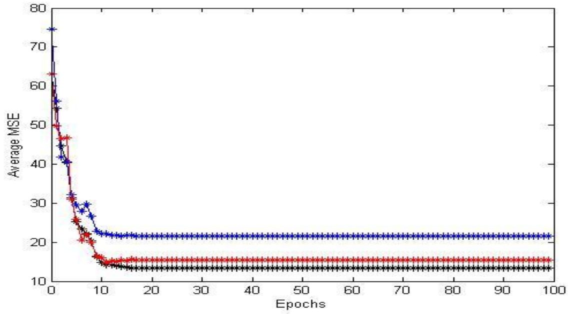
a.MSE vs Time

b.MSE vs epochs

Figure 2. The MSE Hybrid FLANN results vs. time and epochs applied to the iris database



a. MSE vs Time



b. MSE vs epochs

Figure 3. The MSE FLANN based back-propagation results vs. time and epochs applied to the iris database

TABLE II. AVERAGE COMPARATIVE PERFORMANCE OF HFLANN AND FLANN WITH A CONFIDENCE INTERVAL LEVEL OF 5%

| Data bases | Folds | HFLANN | | FLANN BASED BP | |
|---|---|---|---|---|---|
| **IRIS** | *training* | **0,9833** | ±0,0054 | 0,8958 | ±0,0351 |
| | *validation* | **0,9600** | ±0,0303 | 0,9000 | ±0,0228 |
| | *test* | **0,9600** | ±0,0288 | 0,8933 | ±0,0522 |
| | *Time* | 184,3925 | | 142,6253 | |
| **VOTING** | *training* | **0,9787** | ±0,0057 | 0,7796 | ±0,0695 |
| | *validation* | **0,9654** | ±0,0227 | 0,7990 | ±0,0765 |
| | *test* | **0,9469** | ±0,0190 | 0,7829 | ±0,0781 |
| | *Time* | 292,5714 | | 78,4182 | |
| **BREAST** | *training* | **0,9787** | ±0,0022 | 0,9277 | ±0,0192 |
| | *validation* | **0,9699** | ±0,0101 | 0,9342 | ±0,0192 |
| | *test* | **0,9527** | ±0,0139 | 0,9298 | ±0,0179 |
| | *Time* | 211,6272 | | 156,4693 | |
| **PRIMA** | *training* | **0,7906** | ±0,0061 | 0,6865 | ±0,0130 |
| | *validation* | **0,7773** | ±0,0170 | 0,7161 | ±0,0153 |
| | *test* | **0,7501** | ±0,0255 | 0,6536 | ±0,0213 |
| | *Time* | 161,3451 | | 133,8419 | |
| **CREDIT** | *training* | **0,8926** | ±0,0056 | 0,5830 | ±0,0540 |
| | *validation* | **0,8771** | ±0,0264 | 0,6021 | ±0,0444 |
| | *test* | **0,8615** | ±0,0284 | 0,5935 | ±0,0817 |
| | *Time* | 355,2951 | | 100,7891 | |
| **BALANCE** | *training* | **0,9212** | ±0,0036 | 0,6123 | ±0,0157 |
| | *validation* | **90,7400** | ±0,0137 | 0,6454 | ±0,0383 |
| | *test* | **0,9101** | ±0,0104 | 0,6036 | ±0,0367 |
| | *Time* | 314,7218 | | 168,9399 | |
| **WINE** | *training* | **0,9972** | ±0,0036 | 0,9244 | ±0,0549 |
| | *validation* | **0,9833** | ±0,0174 | 0,9379 | ±0,0719 |
| | *test* | **0,9508** | ±0,0329 | 0,9035 | ±0,0399 |
| | *Time* | 173,4038 | | 61,0904 | |
| **BUPA** | *training* | **0,7666** | ±0,0175 | 0,5630 | ±0,0249 |
| | *validation* | **0,7147** | ±0,0395 | 0,6202 | ±0,0276 |
| | *test* | **0,7027** | ±0,0199 | 0,5392 | ±0,0506 |
| | *Time* | 232,4010 | | 42,2137 | |
| **ECOLI** | *training* | **0,8077** | ±0,0157 | 0,6197 | ±0,0559 |

| | | | | | |
|---|---|---|---|---|---|
| | *validation* | **0,7703** | ±0,0342 | 0,6282 | ±0,0630 |
| | *test* | **0,7889** | ±0,0221 | 0,6279 | ±0,0811 |
| | *Time* | 315,8144 | | 255,8217 | |
| GLASS | *training* | **0,7103** | ±0,0181 | 0,3792 | ±0,0419 |
| | *validation* | **0,6901** | ±0,0342 | 0,4328 | ±0,0450 |
| | *Test* | **0,6054** | ±0,0464 | 0,3463 | ±0,0577 |
| | *Time* | 508,9287 | | 174,2889 | |
| ZOO | *training* | **0,8935** | ±0,0310 | 0,4683 | ±0,0789 |
| | *validation* | **0,8977** | ±0,0406 | 0,5005 | ±0,1055 |
| | *test* | **0,8322** | ±0,0422 | 0,4163 | ±0,0757 |
| | *Time* | 193,7053 | | 53,7702 | |

TABLE III. SUMMARY OF THE RESULTS COMPARING THE PERFORMANCE THE HFLANN AGAINST EXISTING WORKS. THE TABLE SHOWS THE REPORTED MEAN CLASSIFICATION ACCURACY OF THE VARIOUS WORKS AND THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

| Evolutionary algorithm | Best Local improvement | Best Global improvement |
|---|---|---|
| **GA** | Crossover | Crossover |
| **DE** | trigonometric mutation scheme | DE/best/1 |
| **PSO** | PSO based genetic algorithms | PSO based Euclidian distance |

TABLE IV. NUMBER OF LOCAL IMPROVEMENT AND GLOBAL IMPROVEMENT OF GENETIC ALGORITHMS, DIFFERENTIAL EVOLUTION AND PARTICLE SWARM

| Methods/ references | Data set | | | |
|---|---|---|---|---|
| | Cancer | Pima | Iris | Balance |
| HFLANN | 0,9527 | **0,7501** | **0,960**0 | **0,9101** |
| C4.5 | 0,947 | 0,7313 | 0,9400 | 0,783 |
| MLP | 0,92 | 0,7330 | 0,9453 | 0,853 |
| RBF | - | 076 | 0,3850 | - |
| SVM | **95,43** | 0,6780 | - | 0,876 |
| K-NN | - | 0,6760 | - | 0,876 |

TABLE V. BEST LOCAL IMPROVEMENT AND BEST GLOBAL IMPROVEMENT OF GENETIC ALGORITHMS, DIFFERENTIAL EVOLUTION AND PSO

| Evolutionary algorithm | Local improvement | Global improvement |
|---|---|---|
| **GA** | 28423 | 662 |
| **DE** | 26148 | 1854 |
| **PSO1** | 104476 | 1246 |

Table V presents best local improvement and best global improvement of genetic algorithms, differential evolution and particle swarm. For differential evolution, trigonometric mutation scheme represent the best local search strategy and DE/best/1 scheme is the best global one. For PSO, basic model based on genetic algorithms represent the best local search strategy, and the PSO based Euclidian distance is the best global one. Comparing genetic operators, we find that crossovers improve results more than mutation.

## V. CONCLUSION

In this paper, a HFLANN was proposed based on three populations based algorithms such genetic algorithms, differential evolution and particle swarm. This classifier shows his ability to converge faster and gives better performance than FLANN based on back-propagation.

When comparing different population based algorithms, we find that PSO is the best local technique for improvement and differential evolution is the best global one. For differential evolution, trigonometric mutation scheme represent the best local search strategy and DE/best/1 scheme is the best global.

For PSO, basic model based on genetic algorithms represent the best local search strategy, and the PSO based Euclidian distance is the best global. Comparing genetic operators, we find that crossovers improve results more than mutation. Following this comparison, we are able to identify best local strategy and best global strategy. Compared to the MLP, FLANN has the advantage to optimize the process without using hidden nodes.

Future works can be addressed to compare other classifiers and others evolutionary algorithms. Others comparison criteria can be used such the needed speed and the robustness of the algorithm. A wrapper approach can be included in the proposed process in order to delete simultaneously irrelevant features over the optimization process.

## REFERENCES

[1] R .Agrawal, T. Imielinski and A. Swami, "Database mining: A performance perspective". IEEE Trans. Knowledge Data Eng., 5, pp. 914-925 (1993).

[2] U. M. Fayyad, Piatetsky-Shapiro, G., and Smyth, P. ,"From data mining to knowledge discovery: An overview". In U. M. Fayyad, G. Piatetsky-Shapiro, & P. Smyth (Eds.), Advances in knowledge discovery and data mining ,pp. 1–34,. Menlo Park, CA: AAAI Press ,1996.

[3] H.P. Kriegel, et al., "Future trends in data mining". Data Mining and Knowledge Discovery, 15(1), 87–97. Netherlands: Springer (2007).

[4] M., James, "Classification Algorithms". Wiley,1985.

[5] L. Breiman, , J.H. Friedman, R.A. Olshen and C.J. Stone, "Classification and Regression Learning". Morgan Kaufman, 1984.

[6] H.P. Kriegel, et al., "Future trends in data mining". Data Mining and Knowledge Discovery, 15(1), 87–97. Netherlands: Springer (2007).

[7] R.O. Duda, Hart PE and Stork D.G., "Pattern classification". Wiley, New York ,2001.

[8] D.E. Goldberg, "Genetic algorithms in search, optimization and machine learning". Morgan Kaufmann ,1989.

[9] G. P., Zhang, "Neural networks for classification", A survey. IEEE Transactions on Systems, Man, Cybernetics-Part C: Application and Reviews, 30(4), pp. 451–461, 2000.

[10] G. P Zhang, "Avoiding pitfalls in neural network research". IEEE Transactions on Systems, Man and Cybernetics-Part C: Applications and Reviews, 37(1), pp. 3–16 , 2007. M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[11] R. Majhi, G. Panda and G. Sahoo, "Development and performance evaluation of FLANN based model for forecasting of stock markets", Expert Systems with Applications 36, pp. 6800–6808,2009.

[12] S. Dehuri and Cho S.B. ,"A hybrid genetic based functional link artificial neural network with a statistical comparison of classifiers over multiple datasets". Neural Comput & Applic (19), pp.317-328. ,2010a.

[13] S. Dehuri and Cho S.B.,"Evolutionarily optimized features in functional link neural network for classification". Expert Systems with Applications (37), pp.4379–4391, 2010b .

[14] L., Fogel, J. Owens and J. Walsh, "Artificial Intelligence through Simulated Evolution". John Wiley, Chichester,1966.

[15] D.E. Goldberg, "Genetic algorithms in search, optimization and machine learning". Morgan Kaufmann ,1989.

[16] J., Holland, "Adaptation in natural and artificial systems". Univ. of Michigan Press, Ann Arbor ,1975.

[17] J. Koza, , "Genetic programming on the programming of computers by means of natural selection", Cambridge MA: MIT Press, 1992.

[18] M. Dorigo, , Stutzle, T.,"Ant Colony Optimization". MIT Press, Cambridge. ,2004.

[19] J., Kennedy Eberhart R., "Particle Swarm Optimization", In Proceedings of IEEE International Conference on Neural Networks, pp. 1942–1948,1995.

[20] R. Storn Kenneth P. "Differential evolution A simple and efficient adaptive scheme for global optimization over continuous spaces". Technical Report TR, pp. 95-012, International Computer Science Institute, Berkeley, CA, 1995.

[21] F. Hui-Y. and J. Lampinen,"A trigonometric mutation approach to differential evolution". In K. C. Giannakoglou, D. T. Tsahalis, J. Periaux, K. D. Papailiou, & T. Fogarty, editors, Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems, pp. 65–70, Athens, Greece. International Center for Numerical Methods in Engineering (Cmine) , 2001.

[22] Y.-H. Pao, S.M. Phillips and D.J. Sobajic, "Neural-net computing and intelligent control systems". Int. J. Contr., 56, pp. 263-289 , 1992.

[23] R. Majhi, G. Panda and G. Sahoo, "Development and performance evaluation of FLANN based model for forecasting of stock markets", Expert Systems with Applications 36, pp. 6800–6808,2009.

[24] C.L. Blake,. and Merz, C.J., 'UCI Repository of machine learning databases', em Irvine, CA: University of California, department of information and Computer Science. Available on-line at : http//www.ics.uci.edu/~mlearn/ML.Repository.html. , 1998.

## AUTHORS PROFILE

**Faissal MILI ,** Ph. Doctor in quantitative methods, A member of Applied Economics and Simulation laboratory, Faculty of Management and Economic Sciences, 5100 Mahdia, Tunisia**.** E-Mail: milisoft@yahoo.fr**.** Phone: 0021620545728

**Manel HAMDI,** Doctoral student in finance, Member of International Finance Group Tunisia, Faculty of Management and Economic Sciences of Tunis, El Manar University, Tunis cedex, C. P. 2092, El Manar Tunisia. E-Mail: mannelhamdi@yahoo.fr. Phone: 0021627220884