

GENETIC ALGORITHMS WITH DIVERSITY MEASURES TO BUILD CLASSIFIER SYSTEMS

Leidys Cabrera Hernández¹*, Alejandro Morales Hernández*, Gladys M. Casas Cardoso*, Yailen Martínez Jiménez*

*Departamento de Computación, Facultad Matemática, Física y Computación, Universidad Central Marta Abreu de Las Villas, Cuba.

ABSTRACT

The combination of classifiers is an active research area of the machine learning and pattern recognition communities. Many theoretical and empirical studies have been published demonstrating the advantages of the paradigm of combination of classifiers over the individual classifiers. When combining classifiers it is important to guarantee the diversity among them [16]. Some statistical measures can be used to estimate how diverse the ensembles of classifiers are, they are called diversity measures.

On the other hand the number of individual classifiers are very big and just a litter group of classifiers can be generated a large number of combinations, so emerge the idea to use one of the meta-heuristics: the genetic algorithms (gas).

Genetic algorithms play a significant role as search technique for handling complex spaces in many fields. They are based on the underlying genetic process in biological organisms and on the natural evolution principles of populations. These algorithms process a population of chromosomes, which represent search space solutions, with three operations: selection, crossover and mutation.

In this paper some diversity measures are presented and one modulation of genetic algorithm using diversity measures is enunciated and implemented in order to obtain, from all possible combinations of a large number of base classifiers, a combination that ensures greater diversity among the chosen classifiers and the best accuracy in multi classifier system. We also present and discuss the results of applying the implemented system in two fields of application. Finally, the general conclusions are exposed.

KEYWORDS: Genetic Algorithms, Diversity Measures, Classifier, Multi Classifiers.

MSC: 68T20.

RESUMEN

La combinación de clasificadores es un área de investigación activa en el aprendizaje reforzado y el reconocimiento de patrones. Muchos estudios teóricos y empíricos han demostrado las ventajas del paradigma de la combinación de clasificadores sobre los clasificadores individuales. Cuando se combinan clasificadores es importante garantizar la diversidad entre ellos. Algunas medidas estadísticas pueden utilizarse para estimar la diversidad de un conjunto de clasificadores, estas se llaman medidas de diversidad.

Por otro lado, el número de clasificadores individuales existentes es muy grande y con sólo un grupo pequeño de clasificadores se puede generar un gran número de combinaciones, entonces surge la idea de utilizar una de las meta-heurísticas existentes: los algoritmos genéticos.

Los algoritmos genéticos juegan un papel importante como técnica de búsqueda para el manejo de espacios complejos en muchos campos. Se basan en el proceso genético subyacente en los organismos biológicos y en los principios de la evolución natural de las poblaciones. Estos algoritmos procesan una población de cromosomas, los cuales representan las soluciones del espacio de búsqueda, esto lo hacen con tres operaciones: selección, cruzamiento y mutación.

En este artículo se presentan algunas medidas de diversidad reportadas en la literatura y se enuncia e implementa una modelación de algoritmos genéticos usando las medidas de diversidad, con el fin de obtener, a partir de todas las combinaciones posibles de un gran número de clasificadores base, una combinación que garantice la mayor diversidad posible entre los clasificadores elegidos y la mejor precisión posible en el sistema multclasificador. Además se presentan y se discuten los resultados del sistema implementado en dos campos de aplicación. Finalmente, se exponen las conclusiones generales.

1. INTRODUCTION

Genetic Algorithms are emerging as tools for solving complex search and optimization problems, as a result of the analysis of adaptive systems in nature. The search and optimization methods have been studied since the early years of computing, ranging from methods based on calculation to enumerative methods, up to random search algorithms. All these methods are analyzed and criticized in terms of robustness, but this does not mean they are not useful; they can be used as a complement to build more robust schemes in order to create hybrid approaches.

The term Genetic Algorithm is used because they simulate the processes of the Darwinian evolution through the use of genetic operators which operate on a population of individuals that evolves from one generation to another.

The development of the whole theory concerning the subject has not only served to achieve an efficient search method but allowed to abstract and rigorously explain the adaptive process in natural systems. Moreover, this has made possible to design artificial systems that include these natural mechanisms [10].

On the other hand, the issue of classification has been widely discussed and continues to develop. Choosing the best classifier depends largely on the problem to be solved, for each case the selected classifier establishes the best decision boundary to

¹ leidysc@uclv.edu.cu, alejandromorales@uclv.cu, gcasas@uclv.edu.cu, yailenm@uclv.edu.cu

separate the classes. In the search for better methods of classification there is a trend to combine several classifiers for the solution of same problem. This is the idea on which the so-called multi classifier systems algorithms are based on. They use several classifiers and combine their outputs with the aim of achieving a better result [25].

Dietteric [6] suggests three reasons why a multi classifier system can be better than a single classifier. The first is statistical, because if each classifier has a hypothesis, the idea of combining these hypotheses, results in a hypothesis that cannot be the best, but at least avoids selecting the worst. The second justification is computational, since some algorithms execute queries that can lead to different local optima: each classifier starts the search from a different point and ends it closes to the optimum. There is an expectation that some combination pathway can lead to a classifier with a better approximation. The last justification is figurative because it is possible that the hypothesis space contains the hypotheses considered not optimal, but the approximation of several decision boundaries can result in a new space outside the initial hypothesis that is closest to the optimum.

There are several ways in which multi classifier systems may be built. There are a number of algorithms developed, some for general problems as bagging and boosting and others for specific problems, but they all have the choice of the fundamental parts of the base classifiers and the choice of how to combine the outputs [4].

The selection of basic classifiers is the first step in building multi classifier systems. Among the variants to combine the base classifiers are the ones used for bagging and boosting, which use the same classification model trained with different case subsets. The first selects random case subsets and the second selects the subsets iteratively based on the result of the previous iteration.

Another variant is used by Stacking, which is used in different classification models trained with the same initial base.

It could be said that these two paradigms are the most general and used in building multi classifier systems. Although which of the two variants is the best has not been demonstrated. Individual multi classifier systems, like simple classifiers, are not intrinsically better than others, but have to be selected on the basis of which gives better results for each specific problem [16].

The diversity between the base classifiers is very important since this will largely depend on the final result of multi classifier systems. Each classifier achieved a percentage of cases classified correct. The more diverse the results of the classifiers are the higher probability to cover a higher percentage of cases, by combining the outputs thereof [14].

Some multi classifier systems ensure diversity using different sets of training bases, but this only works for classifiers that are sensitive to changes, such as decision trees. Others use different sets of features and thus also vary the training base. Others use different base classifiers. In the latter case it is difficult to know when a great diversity is ensured, making it necessary to use some statistical measures that help to determine how diverse they are. Some measures of diversity are described by Kuncheva in [17]. They can be classified as: measures in the form of pairs (pairwise) and measures group (no pairwise). In this paper we will work with the pairwise measures.

2. MEASURES FORMS OF PAIRS (PAIRWISE)

These measures are calculated for pairs of classifiers. Its outputs are binary (0, 1) indicating whether the instance was classified correctly or not. Table 1 shows the results of two classifiers (C_i , C_j) for a given instance, depending on whether or not they were correctly classified. If we consider all instances between the pair of classifiers (C_i , C_j) the following results are obtained, see Table 2:

	C_j correct (1)	C_j incorrect (0)
C_i correct (1)	a	b
C_i incorrect (0)	c	d
a + b + c + d = 1		

Table 1: Binary matrix for one instance

	C_j correct (1)	C_j incorrect (0)
C_i correct (1)	A	B
C_i incorrect (0)	C	D
A + B + C + D = N		

Table 2: Binary matrix for N instances

N is the total number of cases. A set of L classifiers produces L (L-1)/2 pairs of values. To obtain a single result these values are averaged.

Correlation coefficient ρ

The coefficient of correlation [16], is one of the measures for pairs of classifiers, it is calculated as:

$$\rho_{ci,cj} = \frac{A \times D - B \times C}{\sqrt{(A+B) \times (C+D) \times (A+C) \times (B+D)}} \quad (1)$$

A better diversity is obtained for smaller values of ρ . The values of ρ will be in the interval [-1, 1].

Q Statistics

The Q statistic is one of the measures for pairs of classifiers

$$Q_{ci, cj} = \frac{A \times D - B \times C}{A \times D + B \times C} \quad (2)$$

It has been proved that p and Q have the same sign. Also, it can be demonstrated that $|p| \leq |Q|$ [17].

The Measure of Differences

The measure of differences was introduced by Skalak [28], it is the most intuitive measure between a pair of classifiers, and it is equal to the probability that the two classifiers disagree in their predictions. The diversity increases when the value of D increases.

$$D_{ci, cj} = \frac{B+C}{N} \quad (3)$$

The Double-Fault Measure

Another measure to be analyzed is known as double fault measure, which was introduced by Giacinto and Roli [9] and considers the failure of two classifiers simultaneously. This measure is based on the concept that it is more important to know when simultaneous errors are committed, than when both have a correct classification. The diversity increases when the value of DF decreases.

$$D_{ci, cj} = \frac{D}{N} \quad (4)$$

3. GROUP MEASURES

The measures group considers all the classifiers at the same time and calculates a unique value of diversity for the whole set of classifiers.

Entropy

This measure was introduced by Cunningham and Carney [5]:

$$E = \frac{1}{N} \times \frac{2}{L-1} \times \sum_{j=1}^N \min\{(\sum_{i=1}^L Y_{j,i}), (L - \sum_{i=1}^L Y_{j,i})\}, Y_{j,i} \in \{0,1\} \quad (5)$$

Where $Y_{j,i}$ will be 1 if the classifier i was correct in the case j , and 0 otherwise. If E is equal to 0 then there isn't a difference between the classifiers and if E is equal to 1 then there is the most diversity.

Kohavi-Wolpert Variance

The Kohavi-Wolpert Variance was introduced by Kohavi and Wolpert [13], and then Kuncheva and Whitaker presented a modification in [17]. In this measure the diversity is lower if the value of KW is higher.

$$KW = \frac{1}{N \times L^2} \times \sum_{j=1}^N Y(Z_j) \times (L - Y(Z_j)) \text{ donde } Y(Z_j) = \sum_{i=1}^L Y_{i,j} \quad (6)$$

Measurement of Inter-rater Agreement

The Measurement of Inter-rater Agreement was presented in [7]. In this measure the diversity is lower when the k value is higher. The k is calculated by:

$$K = 1 - \frac{\frac{1}{L} \times \sum_{j=1}^N Y(Z_j) \times (L - Y(Z_j))}{N \times (L-1) \times p \times (1-p)} \quad (7)$$

Where the last term is the measure of Kendall concordance and p is the mean of the accurate in the individual classification, which has the following formula:

$$p = \frac{1}{N \times L} \times \sum_{j=1}^N \sum_{i=1}^L Y_{j,i} \quad (8)$$

Coincident Failure Diversity

The Coincident Failure Diversity is enunciated by Partridge y Krzanowski [23], this measure takes into account the instances where all the classifiers coincide.

$$CFD = \begin{cases} 0 & \text{si } p_0 = 1 \\ \frac{1}{1-p_0} \times \sum_{i=1}^L \frac{L-i}{L-1} \times p_i & \text{si } p_0 < 1 \end{cases} \quad (9)$$

This measure has a minimum value of 0 when all the classifiers are correct or incorrect, at the same time. The maximum value is 1 when at least one classifier is incorrect in any random object. In the formula p_i is the probability that $Y=i/L$ and L is the number of classifiers.

Distinctive Failure Diversity

The Distinctive Failure Diversity was also enunciated by Partridge y Krzanowski [22], as an improvement of the previous measure.

$$DFD = \begin{cases} 0 & \text{si } t_i = 0 \\ \sum_{i=1}^L \frac{L-i}{L-1} \times t_i & \text{si } t_i < 0 \end{cases} \quad (10)$$

Where t_i is the number of i fails divided by total distinct fails, and L is the number of classifiers.

4. GENETIC ALGORITHMS

Genetic algorithms are search methods based on the general purpose principles of natural genetics, they are search algorithms based on the mechanisms of natural selection and genetics. Genetic algorithms are an example of a method that exploits the random search guided that has gained popularity in recent years due to their applicability in a wide range of fields and few requirements imposed by the problem [10, 12].

The basic idea is to maintain a population of chromosomes, which represent candidate solutions to the concrete problem; this population evolves over time through a process of competition and controlled variation. Each chromosome in the population has an associated fitness to determine which chromosomes are used to form new ones in the competition process, which is called selection. The new ones are created using genetic operators such as crossover and mutation. GAs have had a great measure of success in search and optimization problems. The reason for a great part of their success is their ability to exploit the information accumulated about an initially unknown search space in order to bias subsequent searches into useful subspaces, i.e., their adaptation. This is their key feature, particularly in large, complex, and poorly understood search spaces, where classical search tools (enumerative, heuristic) are inappropriate, offering a valid approach to problems requiring efficient and effective search techniques.

To use the GAs is necessary to find a structure for representing the possible solutions. Thinking about this issue as the problem of searching in a state space, an instance of this structure represents a point or a state in the search space of all possible solutions. Thus, a data structure in the GA consist of one or more chromosomes (often one), which is commonly represented as a string of bits. Each chromosome is a concatenation of a number of subcomponents called genes. The position of a gene in the chromosome is known as locus alleles. In the string of bits, a gene is a bit, a locus is the position in the string and an allele is its value (0 or 1 if it is a bit).

Fixed length and binary coded strings for the representation of the solutions have dominated the GA research since there are theoretical results that show them to be the most appropriate ones, and as they are easy to implement [11].

In order to optimize the GA structure, a measurement of the quality of each frame in the search space is needed. The fitness function is responsible for this task. In a maximization function, the objective function often acts as the fitness function. The GA usually works with maximization functions, for minimization problems the objective function values can be negated and transferred in order to take positive values, thus producing adaptability [1, 26].

In this meta-heuristic populations of solutions are built, it is stochastic because probabilities are used for taking decisions in the search process, and of course it is bio-inspired because it comes from a natural process.

The simple mechanism of GA is as follows:

- ✓ The simple GA randomly generates a population of n structures (strings, chromosomes or individuals)
- ✓ The population operators act transforming the population. Once the actions of the three operators are completed, we can say that a generational cycle has expired.
- ✓ Then the previous step is repeated while the stopping criterion of the AG is not guaranteed.

The selection operator makes the selection of strings according to their adaptability for the next steps. The crossover operator performs the recombination of genetic material from two parent's strings. The mutation operator, like the natural mutation operator, performs the mutation of a gene within a chromosome.

A probability is associated to each of these operators. The mode of operation of a GA can be summarized as shown in Figure 1. The AG runs for a fixed number of generations or until some stopping criterion is satisfied.

Most experts on this subject agree that the GA can solve the difficulties represented in real life problems that sometimes are insoluble by other methods. The focus of research in GA is robustness: the balance between effectiveness and efficiency needed to survive in many different environments.

In general, Genetic Algorithms have been used in many applications since their emergence, and every year more problems are being solved with this meta-heuristic, some of them from recent years are presented in [1, 2, 3, 8, 15, 18, 19, 24, 26, 27, 29].

5. GENETIC ALGORITHM TO DETECT GOOD ENSEMBLES OF CLASSIFIERS

The Genetic Algorithm configuration depends on the type of problem to solve. If the configuration and representation of all the elements are defined, then it is necessary to define the genetic operators, they are responsible for the evolution each population will have, process in which the expected solution should be found. These operators have to respond to the restrictions of the problem and therefore, they will have to be adapted in some occasions.

In our case the genetic algorithm is presented using diversity measures for combining diverse classifiers and provides the best possible accuracy. The settings of all the parameters of the genetic algorithm and the definition of the goal function are:

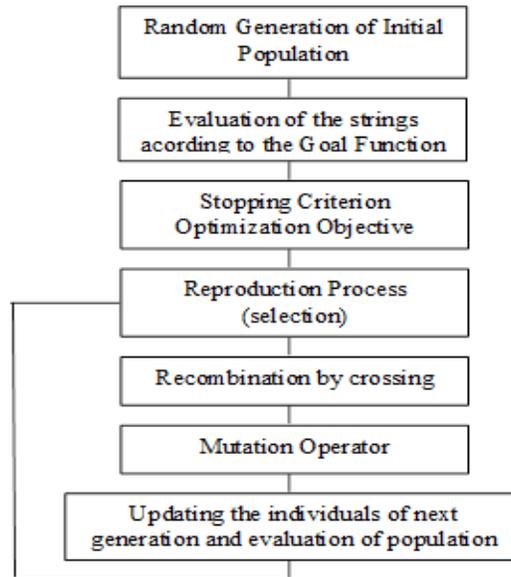


Figure 1: Functional diagram of a Genetic Algorithm

Chromosome configuration

The chromosome will represent the possible solutions of our problem.

Gen: is a binary variable. It takes the value 1 if the classifier belongs to the combination and it takes the value 0 otherwise.

Chromosome: Arrangement of genes representing the set of all the base classifiers that will be used in multiple classifier systems. It is a vector of genes.

The following equation shows the previous aspects:

$$C_x = (g_1, g_2, \dots, g_{Sc}) \text{ where } g_i = \begin{cases} 0 & , \text{if the classifier } i \text{ is not present} \\ 1 & , \text{if the classifier } i \text{ is present} \end{cases} \quad (11)$$

Description of the Goal Function

In our case we want to obtain the better accuracy of multi classifier system together with the better diversity between the classifiers that are used in the multi classifier system. Then the value of f is the sum of the accuracy of the multi classifier system and the result of the diversity measures, according to the configuration of the C_x chromosome.

$$f(C_x) = \text{Accuracy}(C_x) + \text{Diversity}(C_x) \quad (12)$$

Therefore, the goal function in the evolutionary process will be:

$$\max_{0 \leq x \leq P} f(C_x)$$

where P is the population size.

In our problem there can be cases in which the first parameter of the objective function could be small and the value of $f(C_x)$ could be high, because there is a great diversity between the classifiers. As the main goal in this work is to find the combination with higher accuracy in the classification and at the same time find the higher diversity between the classifiers, then another restriction is added. This restriction states that the final result will be the combination where the accuracy of the multiple classifier system surpasses the better accuracy obtained with the individual classifiers, and between them, the combination with more diversity.

Configuration of the population

In the configuration of the population it is necessary to define several elements, for example: the number of individuals in the population and the number of individuals to replace in each iteration.

There are several works related to the influence of the population size in the convergence of the AG. In principle, it is logical to think that the work with small populations has the risk of poorly representing the space of solutions. On the other hand, the populations of great size consume more computational time. In this alternative and as a theoretical work, Goldberg obtained in his investigation that the good size of a population of binary strings, grows exponentially with the longitude of the string [10]. However, in different empiric results many authors suggest sizes of populations as small as 30 individuals.

The population size can be calculated as: $2^{\frac{Sc}{2}}$, where Sc denotes the size of the chromosome. This size was suggested trying to avoid small space solutions or high computational time while analyzing this space.

The initial population will be generated using a hybrid between the random generation and the sown of individuals.

Each chromosome is randomly generated, where each gene takes value 0 or 1 depending on the presence of the classifier; i.e., a random number r is generated, if r is higher than 0.5 the classifier is included and therefore the gene will be 1, otherwise, the classifier is not included and the gene will be 0. After all the chromosomes are generated as explained before, the best individual classifiers are also included in the combination, putting the value of the corresponding gene equal 1. The selection, crossover and mutation operators are used to simulate the genetic recombination and the mechanism of natural selection.

Mutation operator

The implementation of this operator is very simple. The traditional mutation operator is defined: randomly choose a chromosome, randomly choose a gene to mutate and change its status: 0 for 1 or 1 for 0, which means that the inclusion of a classifier changes in the combination. If the resulting chromosome previously exists, choose another mutation point and repeat. If as result of exploration in all mutation points new chromosomes aren't obtained, then another chromosome is selected to mutate. The probability of occurrence of the mutation can be defined by the user.

Crossover operator

In the case of the crossover, it is allowed to select fragments of the genotype of chromosomes that are not very good independently, but when they are mixed, it can be a better solution respect to the previous one. There are several ways to define this operator; in our case we define the classic operator of crossover in a point and the uniform crossover.

In the crossover in a point, two chromosomes are chosen in a random way from the intermediate population, these two chromosomes will act as parents. A position of the gene is randomly chosen and as a result of this crossover two new chromosomes are obtained.

In the uniform crossover, each father has the same probability of contributing with his genes to the only resulting individual. If a randomly generated number is smaller or equal to 0.5, the gene will be taken from the first father; otherwise, it will be taken from the second father.

Once the process of genetic recombination by means of the crossover is concluded, if the new chromosomes already exist in the population then a mutation is performed in order to obtain new and different chromosomes. The probability of occurrence of the crossover can be defined by the user.

Selection operator

In this process an intermediate population of chromosomes is formed, where the previously mentioned operators are applied to obtain a new population with chromosomes that have more quality than the previous ones. For the selection of the chromosomes that will be part of the intermediate population the goal function is used, the objective function evaluated in each one of the individuals will determine their selection to participate in the genetic recombination, the best chromosomes are selected, i.e., the chromosomes with higher value when evaluated using the objective function.

Then the previous operators are applied in this intermediate population to obtain new chromosomes that are added to the initial population. They are added in the initial population because for our problem the combinations that were not selected for the intermediate population sometimes can offer better solutions when they are combined with the new chromosomes and therefore they cannot be deleted. Now the population size is bigger, we denote its size as P' .

Taking into account the characteristics of the problem, in this process one more step is added after recombination, whit the objective of getting a population with the established size. This reduction of the population size is done using a new selection, applying the roulette method, which does not allow the selection of one individual more than once.

In the roulette method the probability used for each chromosome is calculated dividing the result of the goal function for the chromosome by the sum of the goal function of every chromosome in the population with size P' . This is shown in the following formula:

$$p(C_i) = \frac{f(C_i)}{\sum_{i=1}^{P'} f(C_i)} \quad (13)$$

Summarizing, each simple iteration of the GA begins with a population that has size equal to the number previously specified, this population is generated using a hybrid between the random generation and the sown of individuals. The intermediate population is generated by the selection operator, then new chromosomes are generated by the recombination process, they will be added to the initial population and can be taken or not into the new population.

The population will be cleaned of chromosomes that probabilistically take the smaller values in the goal function, until keeping the established size (the roulette method).

The algorithm stops when at least one of the following conditions is true:

- The user specifies to stop when the algorithm finds the first combination that satisfies the conditions and restrictions of the problem.
- The number of generations defined by the user is over.

6. DESIGN OF EXPERIMENTS AND RESULTS

To validate the implemented system taking into account all the aspects mentioned before several experiments were designed. First, three experiments were focused on the validation of the implemented variant of GA and were divided by the number of individual classifiers that were used (6 classifiers, 12 and 18 respectively). The classification models were taken from the WEKA - Waikato Environment for Knowledge Analysis (<http://www.cs.waikato.ac.nz/ml/weka/>) The tables below show the classifiers selected for the experiments:

```

weka.classifiers.bayes.NaiveBayes
weka.classifiers.functions.Logistic -R 1.0E-8 -M -1
weka.classifiers.lazy.IBk -K 1 -W 0 -A weka.core.neighboursearch.LinearNNSearch
-A \ weka.core.EuclideanDistance -R first-last\
weka.classifiers.trees.J48 -C 0.25 -M 2
weka.classifiers.functions.MultilayerPerceptron -L 0.3 -M 0.2 -N 500 -V 0 -S 0
-E 20 -H a
weka.classifiers.trees.ADTree -B 10 -E -3

```

Table 3: Set of 6 classifiers of the first experiment

```

weka.classifiers.bayes.NaiveBayes
weka.classifiers.functions.Logistic -R 1.0E-8 -M -1
weka.classifiers.lazy.IBk -K 1 -W 0 -A weka.core.neighboursearch.LinearNNSearch
-A \ weka.core.EuclideanDistance -R first-last\
weka.classifiers.trees.J48 -C 0.25 -M 2
weka.classifiers.functions.MultilayerPerceptron -L 0.3 -M 0.2 -N 500 -V 0 -S 0
-E 20 -H a
weka.classifiers.trees.ADTree -B 10 -E -3
weka.classifiers.functions.SGD -F 0 -L 0.01 -R 1.0E-4 -E 500
weka.classifiers.trees.RandomTree -K 0 -M 1.0 -S 1
weka.classifiers.functions.SMO -C 1.0 -L 0.001 -P 1.0E-12 -N 0 -V -1 -W 1
-K weka.classifiers.functions.supportVector.PolyKernel -C 250007 -E 1.0
weka.classifiers.lazy.KStar -B 20 -M a
weka.classifiers.functions.VotedPerceptron -I 1 -E 1.0 -S 1 -M 10000
weka.classifiers.bayes.BayesNet -D -Q weka.classifiers.bayes.net.search.local.K2 -
-P 1 -S BAYES -E weka.classifiers.bayes.net.estimate.SimpleEstimator -- -A 0.5

```

Table 4: Set of 12 classifiers of the second experiment

```

weka.classifiers.trees.FT -I 15 -F 0 -M 15 -W 0.0
weka.classifiers.lazy.LWL -U 0 -K -1
-A weka.core.neighboursearch.LinearNNSearch -A \ weka.core.EuclideanDistance
-R first-last\ -W weka.classifiers.trees.DecisionStump
weka.classifiers.trees.RandomForest -I 10 -K 0 -S 1 -num-slots 1
weka.classifiers.lazy.IBk -K 5 -W 0 -A weka.core.neighboursearch.LinearNNSearch
-A \ weka.core.EuclideanDistance -R first-last\
weka.classifiers.functions.SPEgasos -F 0 -L 1.0E-4 -E 500
weka.classifiers.trees.REPTree -M 2 -V 0.001 -N 3 -S 1 -L -1 -I 0.0
weka.classifiers.bayes.NaiveBayes
weka.classifiers.functions.Logistic -R 1.0E-8 -M -1
weka.classifiers.lazy.IBk -K 1 -W 0 -A weka.core.neighboursearch.LinearNNSearch
-A \ weka.core.EuclideanDistance -R first-last\
weka.classifiers.trees.J48 -C 0.25 -M 2
weka.classifiers.functions.MultilayerPerceptron -L 0.3 -M 0.2 -N 500 -V 0 -S 0
-E 20 -H a
weka.classifiers.trees.ADTree -B 10 -E -3
weka.classifiers.functions.SGD -F 0 -L 0.01 -R 1.0E-4 -E 500
weka.classifiers.trees.RandomTree -K 0 -M 1.0 -S 1
weka.classifiers.functions.SMO -C 1.0 -L 0.001 -P 1.0E-12 -N 0 -V -1 -W 1
-K weka.classifiers.functions.supportVector.PolyKernel -C 250007 -E 1.0
weka.classifiers.lazy.KStar -B 20 -M a
weka.classifiers.functions.VotedPerceptron -I 1 -E 1.0 -S 1 -M 10000
weka.classifiers.bayes.BayesNet -D -Q weka.classifiers.bayes.net.search.local.K2 -
-P 1 -S BAYES -E weka.classifiers.bayes.net.estimate.SimpleEstimator -- -A 0.5

```

Table 5: Set of 18 classifiers of the third experiment

As multi classifier system we used Vote , which is included in WEKA too, this multi classifier was selected because its operation is quite simple, it works with the basic idea of combining several individual classifiers and get a single output through a criterion of combination to their output.

The following criteria of combination of the outputs of the individual classifiers were considered:

- Majority Vote: The class that will be assigned is where most of the classifiers agree.
- Average: The class is assigned considering the average of the probabilities of each instance’s classification.
- Product: The class is assigned considering the product of the probabilities of the each instance’s classification.
- Minimum: The class is assigned considering the minimum of the probabilities of the classification of each instance.
- Maximum: The class is assigned considering the maximum probability of the classification of each instance.

The objective of these three experiments was not to propose a set of configuration parameters for the GA, so that in general we can say that they were executed using the classical and uniform crossover operators.

The specific parameters used in the experiments are the following:

- number of generations: 50
- probability of mutation: 0.25
- probability of crossover: 0.75

As evaluation measure we used Percentage Split , taking 66% of the instances for training and the rest for evaluation. The diversity measure selected was Double Fault (DF), because it is one of the most intuitive and simple.

Nine bases from the Repository of Machine Learning Databases from the University of California Irvine-UCIML (<http://archive.ics.uci.edu/ml/>) were used, below are listed some of their features: (see Table 6).

Data Bases	Nominal Features	Numeric Features	Classes	Instances	Distribution by classes
Australian	5	9	2	690	383-307
Breast Cancer Wisconsin	9	0	2	683	444-239
Diabetes	0	8	2	768	500-268
Echocardiogram	1	11	2	132	79-53
Heart Slatlog	0	13	2	270	150-120
Hepatitis	13	6	2	155	32-123
House-votes	16	0	2	435	201-99
German Credit	13	7	2	1000	300-700
Tic Tac Toe	9	0	2	958	626-332

Table 6: Data bases used in experiments

As it was shown, these databases are different: two have numerical features, other three have nominal features and four present a combination of them. They all have two classes. The amount of instances is also different, from bases with 132 cases to bases with 1000 cases.

Figure 2 shows the best accuracy of the classification models used in each experiment and for each database that was used. As we can see, the average accuracy in each group of classifiers was above 85%, resulting significant the accuracy of the bases Breast Cancer Wisconsin, House Votes and Tic Tac Toe with over 95% of cases correctly classified. Usually, the best individual accuracy that the multi classifier should overcome not was constant when the number of classifiers to combine increased.

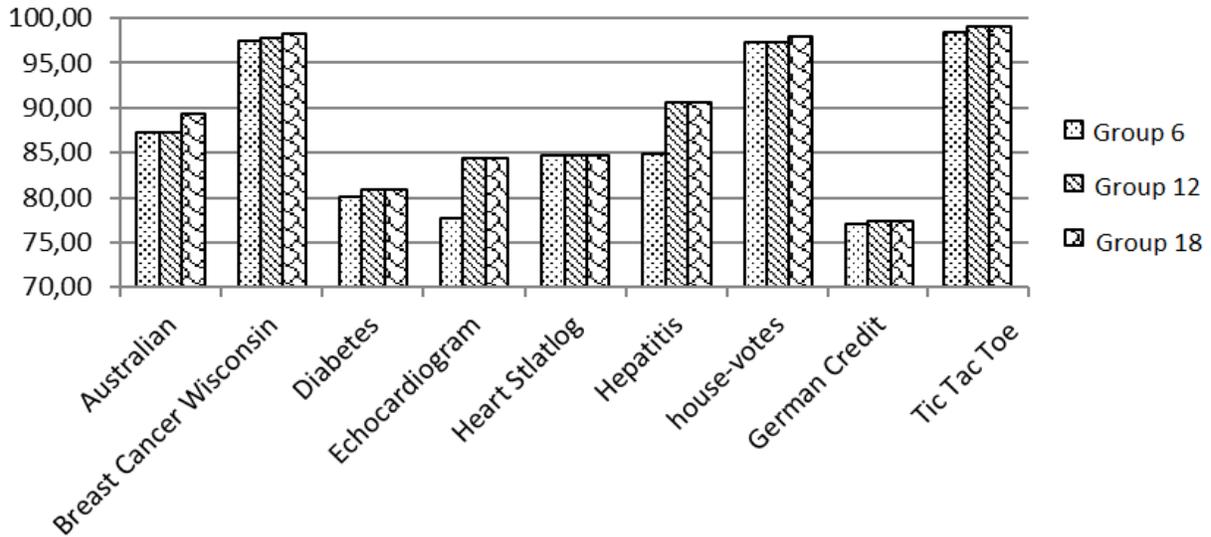
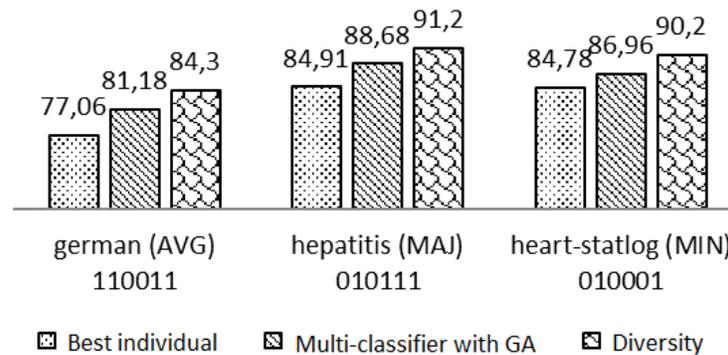
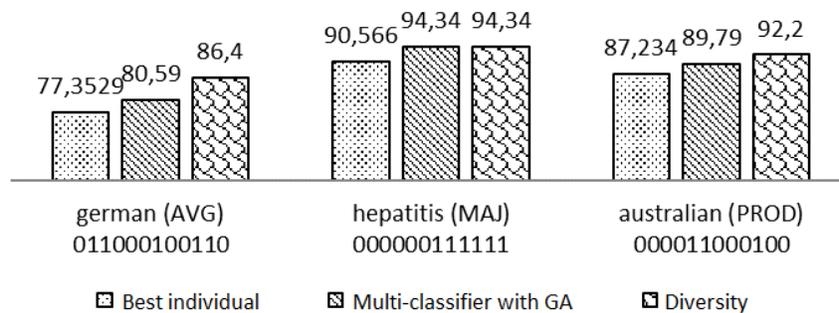


Figure 2: Comparison between the accuracy in the classification of each group of classifiers

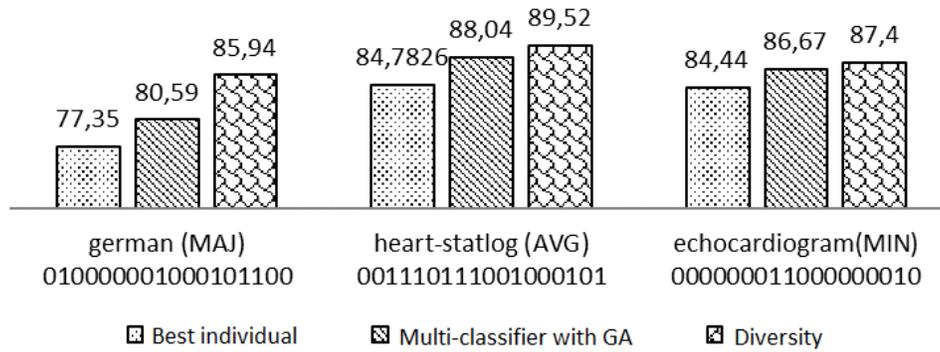
Sometimes several configurations which meet the requirements defined above in the modeling of the genetic algorithm were obtained, that is why representative combinations of each run were selected, taking into account that with a greater accuracy we also get greater diversity in the entire set of solutions found in this run. This way, the most important representative combinations are shown in Figure 3.



a) First experiment with 6 classifiers



b) Second experiment with 12 classifiers



c) Third experiment with 18 classifiers

Figure 3: Best results found with the GA

From the results shown in the comparisons, according to the combination rule of multi classifier and the differences between the best individual accuracy and the accuracy of the formed multi classifier, we determined the combination rule which provided more combinations which accuracy over the best individual.

Figure 4 shows the total of the differences between the accuracy of the multi classifier formed from the combination found, and the best individual accuracy, considering each combination rule of the multi classifier. Note that the combination rules which provided combinations of classifiers with higher differences in individual accuracy were the average of the probabilities and the majority vote.

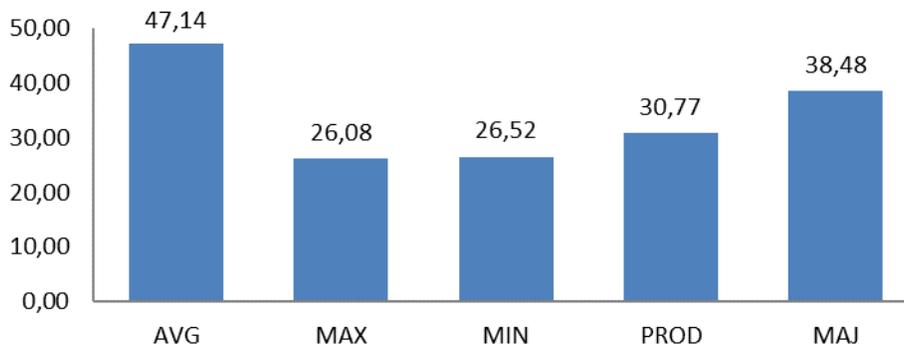


Figure 4: Total of the differences between the best individual accuracy and the multi classifier formed in each combination rule

Also, if we analyze the configuration of each chromosome in the representative combinations, we can detail the results shown in Table 7.

	Group 6	Group 12	Group 18	TOTAL
Total of combinations found	37	38	40	115
<i>From them: have <u>all</u> of the best classifiers</i>	27	18	25	70
<i>From them: have <u>partially</u> the best classifiers</i>	7	11	6	24
<i>From them: don't have <u>any</u> of the best classifiers</i>	3	9	9	21

Table 7: Analysis of the presence of the classifiers of the best accuracy in the representative combinations selected

With this is evident the convenience of the definition of the initial population randomly generating each gene in the chromosomes but forcing to include at least half of the classifiers which accuracy is the best.

Consider a representative slot as a combination obtained after the execution of the GA using a base, and using a combination rule in the multi classifier and with a number of classifiers specified in one of the three experiments.

If you consider that five combination rules were tested in the multi classifier with 10 bases and three sets of classifiers with different sizes, then the processing of these three experiments was to find 150 representative slots as maximum, 50 in each group

of classifiers. Figure 5 shows the behavior of the representative slots (representative combinations of classifiers) found in each of these groups.

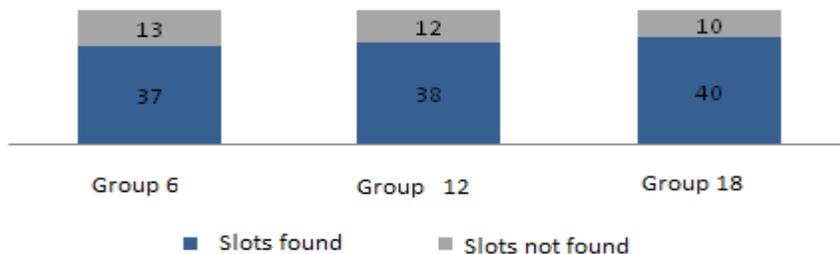


Figure 5: Number of *representative slots* found and not found in the classifier sets defined

From the above, we can see that in the experiments the number of combinations found increased with the number of classifiers, even when the best individual accuracy was not the same in each set or nearly impossible to over (see Figure 2), which supports the relevance of the variant of GA implemented.

As an example, Figures 6a and 6b show a comparison by set of classifiers and the number of representative slots found in the different bases: Diabetes, Echocardiogram and Hepatitis. This also takes into account the accuracy of the multi classifier system built from them. In the rest of the bases used, the slots found were approximately constant.

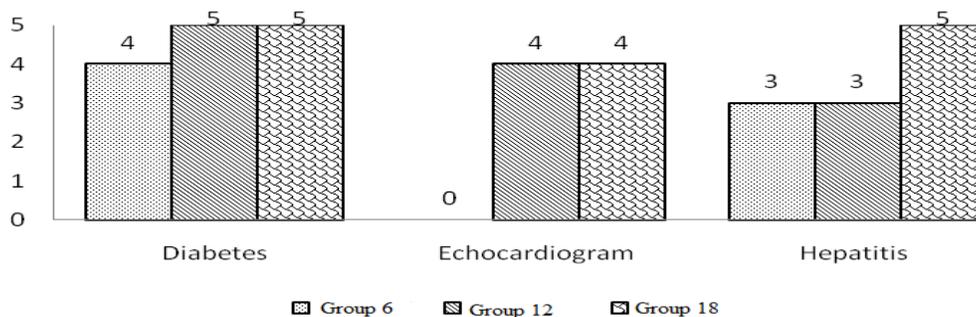


Figure 6a: Behaviour of number of representative slots found

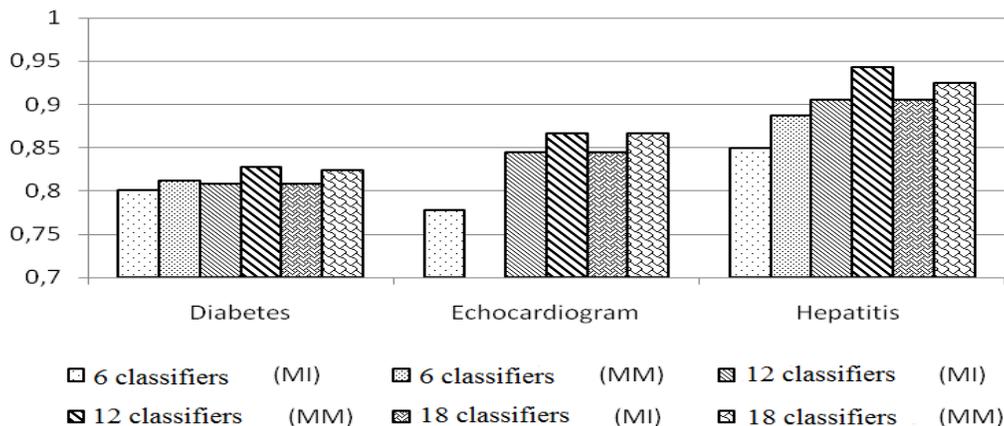


Figure 6b: Accuracy of the multi classifier systems

As it can be seen, there are cases in the experiments in which the set of classifiers did not find any representative slots (Echocardiogram base), or the number found was less than the total (5), but as the number of classifiers varies and also the best accuracy registered, then possible combinations emerge and they can be used in a multi classifier system. This is because the aggregated classifiers were able to classify instances that, when they were combined in multi classifier systems, conducted to correct classification of these instances. However, the best individual classifier of the previous provided a wrong classification. Therefore the accuracy found shows variability in the sets of classifiers. Finally, Figure 7 shows the behavior of the diversity measures and the best multi classifier system formed in each base, taking into account the number of classifiers in the three experiments. Note that apparently there exists a correlation between both variables, confirmed also in the dispersion graphics: b),

c) y d), where we can see that R^2 expresses the quality of the formed lineal function and it is very good. Each graph has accuracy vs. diversity.

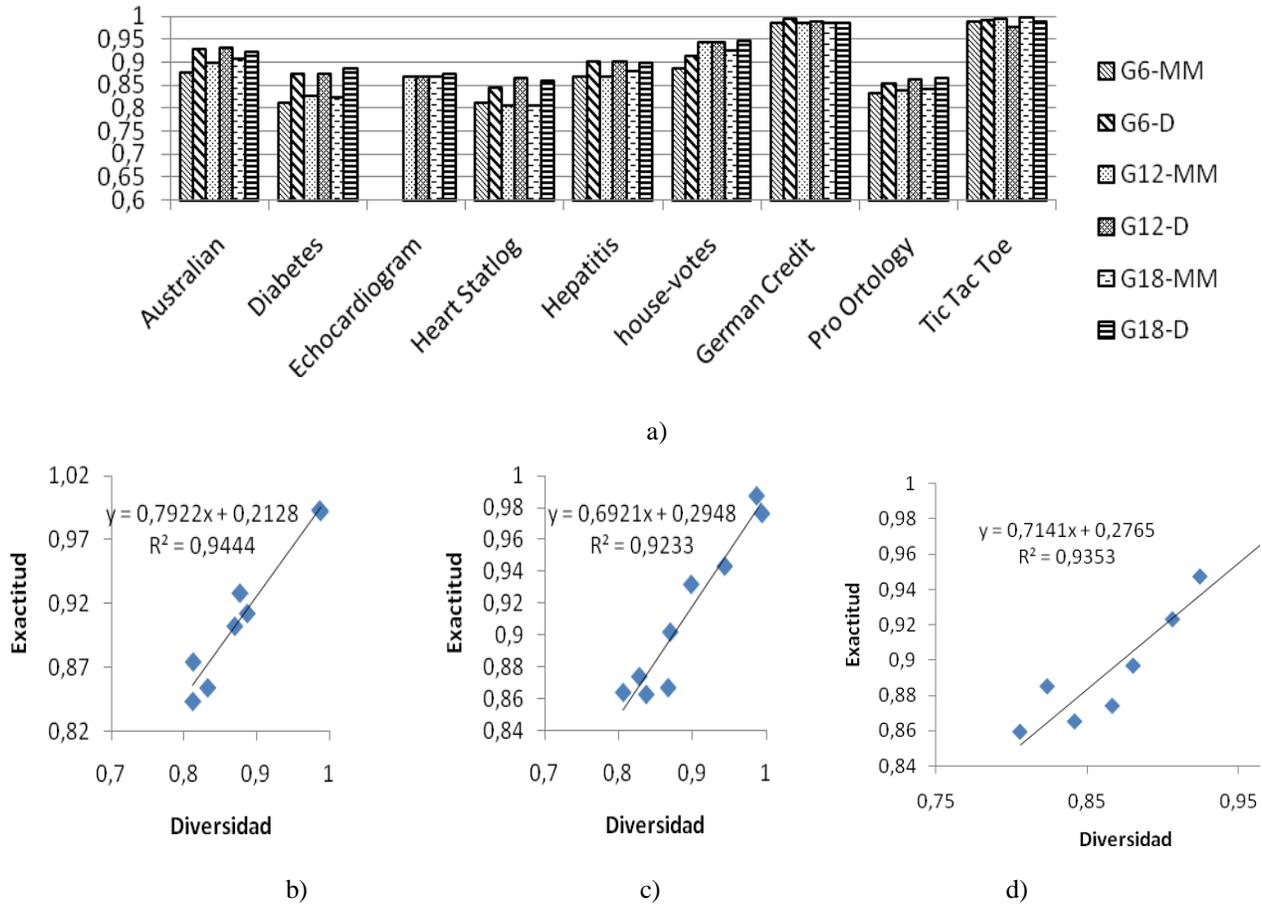


Figure 7: Behaviour of accuracy and diversity in the representative slots, a) Behaviour in each one of the bases b), c) and d) Correlation between the accuracy and the diversity in sets of 6, 12 and 18 classifiers respectively

7. APPLICATION TO PREDICT PROTEIN INTERACTIONS

The problem is to predict protein interactions in a database of *Arabidopsis thaliana*, which was obtained by the Department of Plant Systems Biology², from documents reported in the literature. This database contains information relevant to the interactions of proteins in *Arabidopsis thaliana*: attributes of conserved domains, expression values to calculate Pearson correlation coefficients, information of annotations of GO (Gene Ontology), OG (Orthologous Group), among others.

The data consist of 4314 pairs of proteins, 1438 are examples of true interactions and 2876 are negative examples (or at least dubious). The results reported before demonstrate that the process of identifying simultaneously positive and negative examples is difficult, it is rare to find reports of pairs of proteins non-interacting, especially at large-scale and the negative cases for the learning are not entirely safe.

Eleven features were selected in total, also the special variable named class, which identifies if there is or there is not a protein interaction.

- GO similarity score biological process: average (GO_sim_bp_avg)
- GO similarity score biological process: sum (GO_sim_bp_sum)
- GO similarity score biological process: maximum (GO_sim_bp_max)
- GO similarity score cellular component: average (GO_sim_cc_avg)
- GO similarity score cellular component: sum (GO_sim_cc_sum)
- GO similarity score cellular component: maximum (GO_sim_cc_max)
- Pearson correlation coefficient for micro-array type 1 (PCC_1_dev tissues)
- Pearson correlation coefficient for micro-array type 2 (PCC_2_heterog)
- Domain score 1: number of common domains (domain_match)

² Department of Plant Systems Biology, Flanders Interuniversity Institute for Biotechnology (VIB), Ghent University, Belgium

- Domain score 2: number of common domains/ total number of different domains for the two proteins together (domain_score)
- class (value zero identify no interaction protein, and the value one that there is interaction protein)

A brief statistical description is shown in Table 8.

Input parameters

The multi classifier Vote from WEKA was used together with each combination rule: Average, Majority vote, Product, Maximum and Minimum, as a measure of evaluation we used Percentage Split taking 66% of the instances to train and the rest for evaluation, with DF as diversity measure.

The GA was executed with 0.25 as probability of occurrence of a mutation and 0.75 as probability of occurrence of a crossover, the number of generations for evolution was equal to 50 and the uniform crossover operator was used.

Features	Minimum	Maximum	Media	Std. Deviation
GO_sim_bp_avg	.00	.67	.0082	.02679
GO_sim_bp_sum	.00	1726.00	29.9580	116.75787
GO_sim_bp_max	.00	11.00	.7728	1.96263
GO_sim_cc_avg	.00	2.00	.0199	.09019
GO_sim_cc_sum	.00	1240.00	5.4390	37.63749
GO_sim_cc_max	.00	7.00	.6938	1.56981
PCC_1_devtissues	-.82	1.00	.0635	.33527
PCC_2_heterog	-.80	1.00	.0787	.35708
domain_match	.00	11.00	.3401	1.00913
domain_score	.00	1.00	.1243	.31261

Table 8: Statistical description of the base used

Used Classifiers

The sets of 6, 12 and 18 classifiers defined above in previous experiments were the classifiers used in this application.

Results

The best individual accuracies are observed in Table 9.

	6 classifiers	12 classifiers	18 classifiers
Best individual accuracy	0.82344922	0.82344922	0.825494206

Table 9: Best individual accuracies in sets of 6, 12 and 18 classifiers

As we can see, the individual accuracy increases again with the number of classifiers.

By applying GA to this problem we can obtain the configurations shown in Table 10.

Number of Classifiers	Combination rule of multi classifier					
	AVG			PROD		
	Chromosome	Accuracy	Diversity	Chromosome	Accuracy	Diversity
6	010110	0.82617587	0.83526471	010110	0.82481254	0.83526471
12	111000111110	0.83026585	0.85709417	01011000000	0.82549421	0.84151329
18	111011001000111011	0.83026585	0.85725971	100000000010000100	0.82617587	0.83890025

Number of Classifiers	Combination rule of multi classifier					
	MAJ			MAX		
	Chromosome	Accuracy	Diversity	Chromosome	Accuracy	Diversity
6	011111	0.82685753	0.85576005	010001	0.82413088	0.8459441
12	111100011100	0.82890252	0.85717532	010011000100	0.82481254	0.84333106
18	101110011000001110	0.83162918	0.85798682	01000010010000100	0.82617587	0.83696887

Number of Classifiers	Combination rule of multi classifier		
	MIN		
	Chromosome	Accuracy	Diversity
6	010001	0.82413088	0.845944104
12	010001000000	0.82413088	0.845944104
18			

Table 10: Results

obtained with the GA

Except in one combination rule with the set of 18 classifiers, the remaining combinations accuracies obtained were substantially better than the individual accuracy. In 8 of the 14 combinations found the classifier with the greater accuracy was included and the combination rule with greater differences respect to the individual accuracy was the majority vote.

The diversity performance and accuracy of the multi classifier system can be seen in Figure 6. Note that the diversity in the combinations of classifiers found was relatively high.

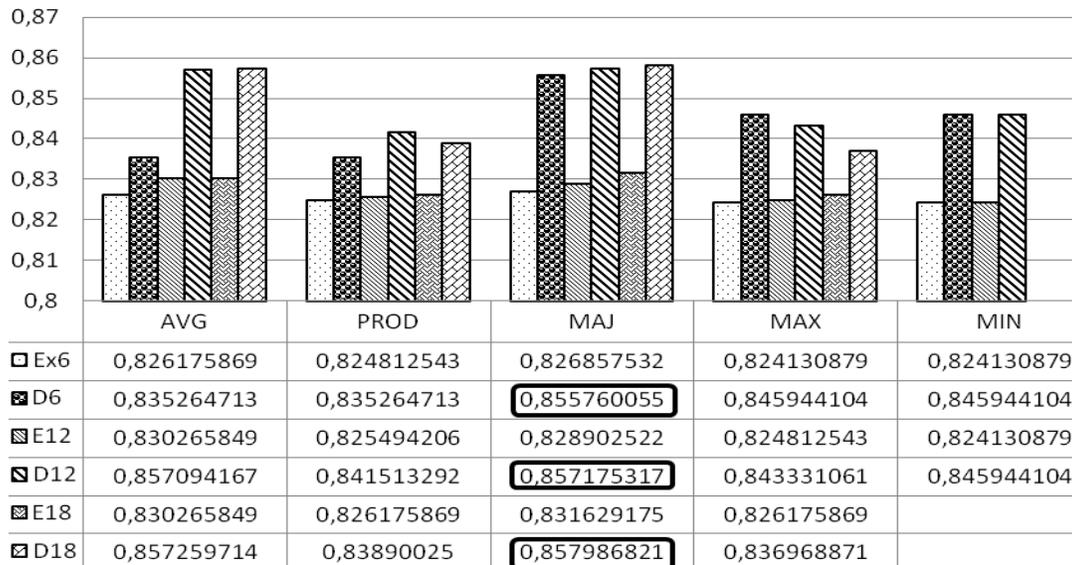


Figure 8: Behaviour of the accuracy in the multi classifier and diversity of the classifiers inside the combination found

8. APPLICATION TO PREDICT SYSTEMIC ARTERIAL HYPERTENSION IN CHILDREN

Nowadays, the concept Systemic Arterial Hypertension is very common in our society and it has been identified as cardiovascular risk factor. However, not everybody thinks in this risk for the children's welfare. The guides to the European Society of Hypertension (ESH) and European Society of Cardiology (ESC) for treatment of this disease, published in 2003 and updated in 2007, did not include, unfortunately, any section on hypertension in children and adolescents [20].

The prevention of cardiovascular diseases should not be limited to adulthood, it should begin in childhood. Diagnosis of hypertension in children is complicated because normal and abnormal values of blood pressure vary with age, sex and height. It has been shown that hypertension in childhood is an independent risk factor for hypertension in adulthood and it is associated with early markers of cardiovascular disease. Considering that morbidity and mortality in long-term are associated with high blood pressure, a major component for the health of children and adolescents is to intervene in time [21].

In this study, the sample consisted of 680 children apparently healthy between 8-12 years of age, of both genders, from 4 primary schools in the city of Santa Clara. The data were supplied by the PROCDEC³ project. We analyzed a set of 31 random variables

³ Project PROCDEC: Projection Electronic Development Center to the Community, Universidad Central Marta Abreu de Las Villas, Santa Clara, Cuba.

used in the diagnosis of cardiovascular risk for predicting the risk of a child to be or not hypertensive. Table 11 shows the key features of the random variables that are discrete and Table 12 shows the basic characteristics of the continuous random variables.

Variables	Identifier	Values	Percent
Sex	Sex	Female	50.4
		Male	49.6
Skin color	Skin	white	81.5
		No	18.5
Diagnostic	Newdiag	yes	57.5
		No	42.5

Table 11: Discret predictor variables

Variables	Identifier	Minimum	Maximum
Age	EdadA	8	11
Current weight (en kg)	PesoKg	20.5	75.1
Size(en cm)	Talla	113	191
Waist circumference (en cm)	CCintura	37	182
Hip circumference (en cm)	CCadera	30	106
Waist hip index	icc	0.58	2.5
TA lower limb Systolic	TSMI	11	150
TA Lower Limb Diastolic	TDMI	20	90
TA Upper Limb Systolic	T1dBS	81	150
TA Upper Limb Diastolic	T1dBD	6	99
TA Systolic 5 min	T1d5S	73	152
TA Diastolic 5 min	T1d5D	46	95
TA Systolic 10 min (before 15 min)	T1d10S	11	149
TA Diastolic 10 min (before 15 min)	T1d10D	45	94
TA 1 ^{er} day, Systolic Average	T1SMedia	77.33	146
TA 1 ^{er} day, Diastolic Average	T1DMedia	43.33	88.67
TA 2 ^{do} day, Basal Systolic	T2dBS	82	150
TA 2 ^{do} day Basal Diastolic	T2dBD	17	94
TA 2 ^{do} day Systolic P. Sustained weight	T2dPPSS	78	174
TA 2 ^{do} day Diastolic P. Sustained weight	T2dPPSD	48	116
Arterial Pressure Average 2d	PAM2d	58	129.67
TA 3 ^{er} day Basal Systolic	T3dBS	81	144
TA 3 ^{er} day Basal Diastolic	T3dBD	40	110
TA 3 ^{er} day Systolic P. Sustained weight.	T3dPPSS	83	160
TA 3 ^{er} day Diastolic P. Sustained weight.	T3dPPSD	13	112
Average of Arterial Pressure 3d	PAM3d	53.33	120
Índice of corporal weight	imc	8.77	255.19

Table 12: Continuous predictor variables

Input parameters

The multi classifier Vote from WEKA was used together with each combination rule: Average, Majority vote, Product,

Maximum and Minimum, as a measure of evaluation we used Percentage Split taking 66% of the instances to train and the rest for evaluation, with DF as diversity measure. The GA was executed with 0.25 as probability of occurrence of a mutation and 0.75 as probability of occurrence of a crossover, the number of generations for evolution was equal to 50 and the uniform crossover operator was used.

Used Classifiers

The sets of 6, 12 and 18 classifiers defined above in previous experiments were the classifiers used in this application.

Results

The best individual accuracies are observed in Table 13.

	6 classifiers	12 classifiers	18 classifiers
Best individual accuracy	0.9004	0.9091	0.9134

Table 13: Best individual accuracies in sets of 6, 12 and 18 classifiers

It can be seen that the individual accuracy increases again with the number of classifiers. By applying GA to this problem we can obtain the configurations shown in Table 14.

Number of Classifiers	Combination rule of multi classifier					
	AVG			PROD		
	Chromosome	Accuracy	Diversity	Chromosome	Accuracy	Diversity
6	0100 <i>I</i> 0	0.9081	0.9089	0101 <i>I</i> 0	0.9072	0.9080
12	111000 <i>I</i> 11110	0.9288	0.9319	010111000000	0.9129	0.9153
18	11101100100011 <i>I</i> 011	<u>0.9480</u>	0.9496	10000000010100100	0.9290	0.9299

Number of Classifiers	Combination rule of multi classifier					
	MAJ			MAX		
	Chromosome	Accuracy	Diversity	Chromosome	Accuracy	Diversity
6	0111 <i>I</i> 1	0.9074	0.9085	010001	0.9072	0.9082
12	101100 <i>I</i> 11100	0.9281	0.9309	010011000100	0.9129	0.9150
18	00100100010010 <i>I</i> 100	0.9468	0.9480	01000010010000100	0.9290	0.9299

Number of Classifiers	Combination rule of multi classifier		
	MIN		
	Chromosome	Accuracy	Diversity
6	0100 <i>I</i> 1	0.9072	0.9082
12	010001 <i>I</i> 00000	0.9174	0.9184
18	001001000100100100	0.9272	0.9289

Table 14: Results obtained with the GA.

The accuracy of the combinations obtained was substantially better than the individual accuracy. In 9 of the 15 combinations found the classifier with the greater accuracy was included, and the combination rule with greater differences respect to the individual accuracy was the average. Also, we can see that the second best results found in the combination rules were using majority vote.

The diversity performance and accuracy of the multi classifier system can be seen in Figure 7. Note that the diversity in the found combinations of classifiers was relatively high, same to the first application.

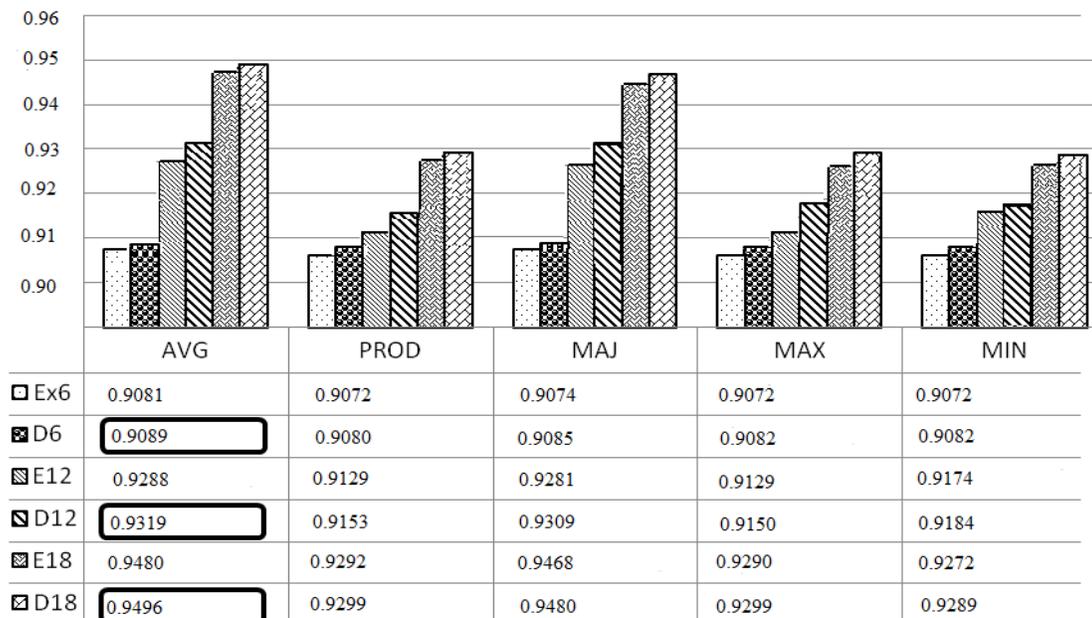


Figure 9: Behaviour of the accuracy in the multi classifier and diversity of the classifiers inside the combination found

Figure 10 shows a comparison of results between the accuracy of individual classifiers and the multi classifier system found. Note that the multi classifier is better.

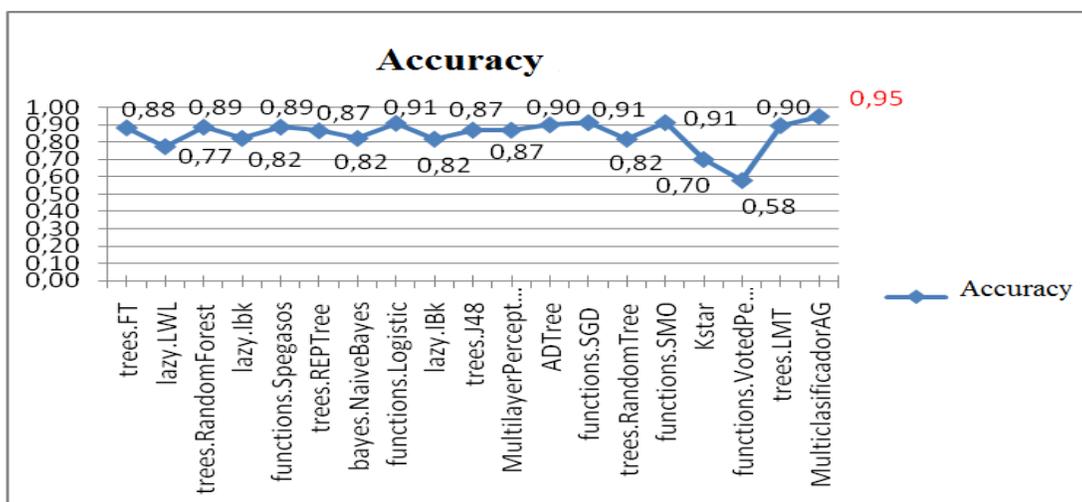


Figure 10: Accuracy of 18 models used and multi classifier obtained in this application

9. DISCUSSION SECTION

Several experiments were designed and conducted in order to validate the GA variant presented in this paper. The results of these experiments demonstrated that the implemented GA is able to find a combination of classifiers which inclusion in the multi classifier system achieves accuracy greater than the individual best classification accuracy. It is also shown how the classifiers included in the combinations found are diverse from each other.

Specifically, two different applications are used, in the first one we worked with a database that describes the interaction of proteins in *Arabidopsis thaliana*, which shows that the best (or better) individual classifiers were completely or partly included in the combinations that yielded the best results. Also, the probability of finding a combination of classifiers that meets the requirements of the problem increased with the number of classifiers that were combined. Moreover, the combination rule where the best results were found was the majority vote. In general 18 classifiers were applied and their individual results showed 82% as best accuracy. Using the GA we obtained a combination of these classifiers which accuracy was better (83%) than the one reported by the individual classification.

In the second application, we worked with a database that describes the necessary characteristics to predict the presence of *hypertension in children*; it is noted again that the best (or better) individual classifiers are completely or partly included in the combinations that yielded the best results. Again, the probability of finding a combination of classifiers that meets the

requirements problem increases with the number of classifiers that are being combined. In this case the combination rule that yielded the best results was the average although the results using majority vote were very close. In general, the same 18 classifiers were applied, and their individual results show 91% as best accuracy. Using the GA we obtained a combination of these classifiers which accuracy was better (94%) than the individual classification.

The same experiment was applied twice in different applications, where it was shown that the best results in a combination of classifiers are achieved when the best of the individual classifiers is included. Also, if there exist more classifiers to combine, the probability of finding a better accuracy increases as a result of some possible combinations. Moreover, the combination rules that best perform are the majority vote and the average.

10. CONCLUSIONS

In this paper we presented a novel technique using Genetic Algorithms to find a good combination of diverse classifiers. The objective function in the genetic algorithm involves the classification accuracy of the multi classifier and the results of the diversity of the base classifiers.

Several experiments were performed, which showed that in the combinations found by the genetic algorithm, as solutions, the best (or better) individual classifiers are completely or partly included.

Also, in these experiments it was shown that the probability of finding a combination of classifiers that meets the requirements of the problem, increases with the number of classifiers that are being combined.

Finally, two different applications are shown. The results obtained in both are similar, where the combinations of classifiers that outperform the best individual accuracy were found, and these classifiers are relatively diverse from each other. All the aspects mentioned before validate the model proposed with genetic algorithms using diversity measures.

RECEIVED NOVEMBER, 2014
REVISED MARCH, 2015

REFERENCES

- [1] ANTONISSE, J. (1989): A new interpretation of schema notation that overturns the binary encoding constraint . In **Proc. of the Third International Conference on Genetic Algorithms**, George Mason University, USA, 86-91.
- [2] AWAD, A. R. and ABOUL-ELA, M. T., (2011): Optimal Removal of Heavy Metals Pollutants from Groundwater Using a Real Genetic Algorithm and Finite-Difference Method . **Journal of Computing in Civil Engineering**, 522-533.
- [3] BASU, M. (2014): Fuel constrained economic emission dispatch using nondominated sorting genetic algorithm-II . **Energy**, Vol 78, Pages 649–664.
- [4] BONET, I. (2008): **Modelo para la clasificación de secuencias, en problemas de la bioinformática, usando técnicas de inteligencia artificial** . Tesis Doctoral , Universidad Central Martha Abreu de las Villas, Santa Clara.
- [5] CUNNINGHAM, P. and CARNEY, J. (2000): **Diversity versus Quality in Classification Ensembles Based on Feature Selection, in Machine Learning** . ECML , R. López de Mántaras and E. Plaza, Springer Berlin / Heidelberg.,
- [6] DIETTERICH, T. G. (2000): **Ensemble methods in machine learning. Multiple Classifier Systems**. Berlin: Springer-Verlag Berlin.
- [7] FLEISS, J. L. (1981): **Statistical Methods for Rates and Proportions**. John Wiley & Sons.
- [8] FRANS, R. and ARFIADI, Y. (2014): Sizing, Shape, and Topology Optimizations of Roof Trusses Using Hybrid Genetic Algorithms . **Procedia Engineering**, 10.1016/j.proeng.2014.12.178, 185-195.
- [9] GIACINTO, G. and ROLI, F. (2001): Design of effective neural network ensembles for image classification purposes , **Image Vision and Computing Journal**, 19, 699-707.
- [10] GOLDBERG, D. E. (1989): **Genetic algorithms in search, optimization, and machine learning** . vol. 412: Addison-Wesley Reading Menlo Park.
- [11] GOLDBERG, D. E. (1990): Real-coded genetic algorithms, virtual alphabets, and blocking . **Urbana**, 51, 61801.
- [12] HOLLAND, J. H. (1975): **Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence** . U Michigan Press.
- [13] KOHAVI, R. and WOLPERT, D. H. (1996): Bias Plus Variance Decomposition for Zero-One Loss Functions in Machine Learning. **Proceedings of the Thirteenth International Conference**.
- [14] KRAWCZYK, B. and WOŹNIAK, M. (2014): Diversity measures for one-class classifier ensembles . **Neurocomputing**.126, 36–44.
- [15] KUMPHON, B. (2013): Genetic Algorithms for Multi-objective Optimization: Application to a Multi-reservoir System in the Chi River Basin, Thailand . **Water Resources Management**, 10.1007/s11269-013-0416-y, 4369-4378.
- [16] KUNCHEVA, L. I., (2004): **Diversity in Classifier Ensembles, Combining Pattern Classifiers: Methods and Algorithms** . Wiley Interscience, N. York .
- [17] KUNCHEVA, L. I. and WHITAKER, C. J., (2003): Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy . **Machine Learning**., 51, 181-207.
- [18] LEE, S.-Y. (2014): An Advanced Coupled Genetic Algorithm for Identifying Unknown Moving Loads on Bridge Decks , **Mathematical Problems in Engineering**, vol 2014, 1-11.
- [19] LI, D., CHEN, S, and HUANG, H. (2014): Improved genetic algorithm with two-level approximation for truss topology optimization. **Structural and Multidisciplinary Optimization**, 10.1007/s00158-013-1012-8, 795-814.

- [20] MANCIA, G., et al. (2007): Guidelines for the management of arterial hypertension: The Task Force for the Management of the Arterial Hypertension of the European Society of Hypertension (ESH) and the European Society of Cardiology (ESC) . **European Heart Journal, J Hypertens.** 25, 1462-1536.
- [21] ORTIGADO, A. (2011): **Hipertensión arterial sistémica. Tratado de Pediatría Extrahospitalaria** . Ergon, Madrid.
- [22] PARTRIDGE, D. and KRZANOWSKI, W., (1997):Distinct failure diversity in multiversion software , **Res. Rep.**, 348.
- [23] PARTRIDGE, D. and KRZANOWSKI, W., (1997): Software diversity: practical statistics for its measurement and exploitation. **Information and Software Technology.** 39, 707-717.
- [24] PENG, S. Z., WANG, Y., KHAN, S., RANA, T., and LUO, Y. F. (2012): A simplified multi-objective genetic algorithm optimization model for canal scheduling. **Irrigation and Drainage**, 10.1002/ird.654, 294-305.
- [25] POLIKAR, R. (2006): Ensemble based systems in decision making . **IEEE Circuits and Systems Magazine**, 6, 21-45.
- [26] RADCLIFFE, N. J. (1992): Non-Linear Genetic Representations in Parallel Problem Solving from Nature. (**PPSN**),. 259-268.
- [27] RANI, D., JAIN, S., SRIVASTAVA, D., and PERUMAL, M. (2013): 3 Genetic Algorithms and Their Applications to Water Resources Systems. **Metaheuristics in Water, Geotechnical and Transport Engineering**, Newnes, 43-78.
- [28] SKALAK, D. B. (1996): The Sources of Increased Accuracy for Two Proposed Boosting Algorithms , Proc. American Association for Arti Intelligence. **AAAI-96, Integrating Multiple Learned Models Workshop**, 120-125.
- [29] WIKAISUKSAKUL, S. (2014): A multi-objective genetic algorithm with fuzzy c-means for automatic data clustering . **Applied Soft Computing**, 24, 679-691.