Int. J. Mach. Learn. & Cyber. (2010) 1:75–87
DOI 10.1007/s13042-010-0004-x

ORIGINAL ARTICLE

Genetic Algorithm-Neural Network (GANN): a study of neural network activation functions and depth of genetic algorithm search applied to feature selection

Dong Ling Tong · Robert Mintram

Received: 30 March 2010 / Accepted: 25 August 2010 / Published online: 10 September 2010

Abstract Hybrid genetic algorithms (GA) and artificial neural networks (ANN) are not new in the machine learning culture. Such hybrid systems have been shown to be very successful in classification and prediction problems. However, little attention has been focused on this architecture as a feature selection method and the consequent significance of the ANN activation function and the number of GA evaluations on the feature selection performance. The activation function is one of the core components of the ANN architecture and influences the learning and generalization capability of the network. Meanwhile the GA searches for an optimal ANN classifier given a set of chromosomes selected from those available. The objective of the GA is to combine the search for optimum chromosome choices with that of finding an optimum classifier for each choice. The process operates as a form of co-evolution with the eventual objective of finding an optimum chromosome selection rather than an optimum classifier. The selection of an optimum chromosome set is referred to in this paper as feature selection. Quantitative comparisons of four of the most commonly used ANN activation functions against ten GA evaluation step counts and three population sizes are presented. These studies employ four data sets with high dimension and low significant datum instances. That is to say that each datum has a high attribute count and the unusual or abnormal data are sparse within the data set. Results suggest that the hyperbolic tangent (tanh) activation function outperforms other common activation functions by extracting a smaller, but more significant feature set. Furthermore, it was found that fitness evaluation sizes ranging from 20,000 to 40,000 within populations ranging from 200 to 300, deliver optimum feature selection capability. Again, optimum in this sense meaning a smaller but more significant feature set.

Keywords Genetic Algorithm (GA) · Artificial Neural Network (ANN) · Activation function · GA evaluation · GA population · Feature selection

1 Introduction

Hybrid genetic algorithms and neural networks have been employed extensively within prediction and classification applications. A typical combination of is to use an ANN as the prime modelling tool and a GA to optimise the ANN weights, or in some cases the ANN topology. For instance, Beiko and Charlebois [1] used the GA to identify the best combinations of ANN topology for DNA sequence classification. Karzynski et al. [7] used GAs to optimise both the architecture and the weights of ANN for multiclass microarray classification. Taheri and Mohebbi [15] used the GA to fine-tune ANN parameters, including the number of nodes in the hidden layers, the momentum and the learning rates. Zorić and Pandžić [17] utilised the GA to find the near optimal ANN topology in the real-time lip synchronisation classification. Theoretical developments have been made in ANNs and GAs with recent research focusing on the use of ANNs to model the GA fitness function. For instance, Cho et al. [4] used the ANN classification results as the GA fitness function in the cDNA microarray prediction. Bevilacqua et al. [2] and Lin et al.

D. L. Tong (✉)
The John van Geest Cancer Research Centre, School of Science and Technology, Nottingham Trent University, Clifton Lane, Nottingham NG11 8NS, UK
e-mail: dong.tong@ntu.ac.uk

R. Mintram
Newcastle University, Newcastle upon Tyne NE1 7RU, UK

[9] applied the error rates returned by ANNs to determine the fitness of GAs in the cancer classification.

We recognize two issues associated with hybrid systems. First, most hybrid systems emphasize effective generalisation capability, i.e. classification, rather than effective feature selection. Sometimes data preprocessing, such as filtering, imputation and normalisation is performed on large and high dimensional data sets prior to classification. The intention being to remove undesirable data characteristics with the idea of ensuring data integrity and improving classification performance.

A potential consequence of data homogenisation is the equalisation of features within the data. What was originally a primary classification feature may become of equal significance to secondary and less significant features. It is certain that care must be taken over this process in order to avoid feature equalisation.

For instance, Verma and Zhang [16] transformed all the feature values to be positive and then normalised these values to the range of 0–1 prior to digital mammogram classification for breast cancer using hybrid GA/ANN. They reported the highest classification rate of 85.0% with five most significant features of a digital mammogram for microcalcification classification. Ramasubramanian and Kannan [11] converted the alphabetic values in the data set to an appropriate numeric value and then normalised each numerical value in the data set to the range of 0.05–0.95 for forecasting the instruction pattern on the database security. They commented that the hybrid GA/ANN model yields better results than the standard ANN model with prefect discrimination on a ROC (Receiver Operating Characteristic) plot.

Second, there is no clear explanation of the architecture of these hybrid systems. Many studies did not provide fundamental information on the configuration of the ANN and the GA, such as the number of nodes in the ANN, the ANN activation function, the ANN acceleration parameter (momentum, learning rate), the GA population size, the number of chromosomes in a population and GA evaluation operator (selection, crossover, mutation). Inaccurate specification of these parameters may lead to an inaccurate classification result. Furthermore and significantly, the reproduction of the results is not possible. For instance, Zorić and Pandžić [17] omitted to completely describe the GA configuration and the number of hidden layers and hidden nodes in their hybrid system. Bevilacqua et al. [2] did not provide information on the GA population size, selection, crossover and evaluation size that was used to select significant features for breast metastasis recurrence classification.

In spite of the importance of the ANN activation function and the GA evaluation, they have not been systematically investigated. Several studies have been performed on the ANN activation function, e.g. Shenouda [14] conducted a quantitative comparison on the four ANN activation functions including linear, sigmoid, hyperbolic tangent and Gaussian, over ten different data sets. However, the integration of these parameters has not been investigated in the hybrid GA/ANN literature.

Therefore, this paper compares the performance of four commonly used ANN activation functions, binary sigmoid, linear, hyperbolic tangent and threshold, in three population sizes and ten GA evaluation depths with respect to feature selection over four high dimensional data sets. For reference we refer to the hybrid architecture as Genetic Algorithm-Neural Network (GANN). Specifically, this paper does not attempt a general comparison of feature extraction proficiency among a selection of commonly used algorithms. Rather it focuses on the empirical performance and behaviour of the GANN algorithm applied to a collection of real and synthetic data sets.

For the remainder of this paper we first describe the GANN method and its parameters. We then describe the data sets and the comparison results. We conclude with a discussion of the results.

2 Genetic Algorithm-Neural Network (GANN)

We consider the GANN procedure as a form of co-evolution of two distinct objectives. In the first case, given a complex high dimensional data set it is desirable to find a classifier that will correctly discriminate between the classes within data. It is apparent in this mode that the objective is to find a classifier. However, because the data is of high dimension it is also desirable to find subsets of the datum elements that enable an accurate classification to be performed. In other words, the idea is to find a feature set obtainable from the original data that will do this job, i.e. feature selection for classification. Naturally, the features that could be selected may and often will depend on the classes into which the data is to be divided. There are many potential features, possibly as many as the dimension of the data. So we conclude that there is a connection between the objectives of finding an optimal classifier and that of extracting a sufficient feature set for the classification to be performed.

In the GANN model we create an architecture that can perform both objectives. But there is a tension here between the capability of the classifier and the suitability of the selected feature set. Ultimately it could be expected that a GA might, if appropriately configured, satisfy both objectives. However in the first instance we recognize feature selection as a primary objective. So for the GANN algorithm, the derived ANN is an artifact of the process and is discarded.

The principal objective is to find a feature set that given a basic ANN classifier will effectively classify the data. The presumption is, and this is a major assumption of our model, that the feature set will actually be the feature set that in some sense correctly acts to discriminate between the classes. That is to say, that by deliberately not focusing on the quality of the ANN classifier then the selected feature set will be closer to the true discriminating feature set for the given classes. This point is worth emphasising. We deliberately use very simple ANN models to avoid the possibility of a sophisticated ANN architecture correctly discriminating between the data classes by using a less than optimal feature set. The capability of the net making up for the inadequacy of the feature set. Since the prime objective is to extract an optimal feature set then we use ANNs with what we assume will be the basic minimum discrimination capability.

Thus we suppose, ensuring that the selected feature set will be optimal. We recognize clearly that this is a significant assumption of our model but we believe it is reasonable and thus acceptable. Very briefly, the genetic algorithm is tasked with finding a neural network together with a feature set that will correctly discriminate the classes from the training data set. Each element of the GA population will contain the ANN configuration and an indicator of the feature set to be selected. How well the ANN performs using the selected feature set determines the fitness of the member of the population.

We now describe in more detail the functionality of the GANN algorithm. Table 1 summarises the GANN parameters.

Table 1 Summary of the GANN parameters

Parameter	Setting
Population initialisation	
Population size	{100, 200, 300}
Chromosome size	10 features
Chromosome encoding	Real-number representation
Fitness computation	
Fitness function	The total number of correctly labelled instances
Selection	Tournament, tournament size = 2
ANN architecture	10-5-O, where O is ranges from 2-4
ANN size	67-79 nodes including 7-9 bias nodes
ANN learning algorithm	Feedforward
ANN activation function	{Binary sigmoid, Linear, Tanh, threshold}
Fitness evaluation	
Crossover operator	Single-point, $P_c = 0{:}5$
Mutation operator	$P_m = 0{:}1$
Elitism strategy	Retain N-1 chromosomes in the population, where N is the total number of chromosomes in the population
Termination criteria	
Evaluation size	{5,000, 10,000, 15,000, 20,000, 25,000, 30,000, 35,000, 40,000, 45,000, 50,000}
Whole cycle repeat	5,000

2.1 Population initialisation

Population initialisation is essentially random. For the ANN aspect of the chromosome (i.e. the selected feature set to construct a member of the population) random weights are generated. Reasonable values between 0 and 1 are chosen. Each index of the feature to be used from the potential range is also initialised at random. For example if there are potential 10,000 element vectors from which 10 are to be chosen to form a feature set then each of the 10 elements will be initialised to a random number between 1 and 10,000. No care is taken to eliminate duplicates since it is assumes that the GA will later work to find an optimum set. Maximum freedom to find this set must be permitted. Duplicate selections may indicate merely that there is a stronger preference for this element.

In the principal experiments underpinning this study, population sizes of 100, 200 and 300 were evaluated. Larger population sizes, i.e. greater than 300 were not considered because it was found that there was no difference in system performance (i.e. median and selected features) at these sizes. This conclusion was based on trial experiments using population sizes varying from 100 to 600 with a fitness evaluation size of 20,000. In this case two simulated data sets were created with each containing 30 predefined features and 100 samples. The summary of the trial results follows in Table 2.

The trial results show that convergence begins when the population size exceeded 200, although for binary class data set, all the 30 predefined features were selected in the population size 100. There was no difference in system performance (i.e. median and selected features) when the population size exceeding 300. There was a marginal difference in the fitness accuracy for population size, varying from 300 to 600 for binary class data set and significant deficiency in the system performance for multiclass data set when the population size exceeding 400.

2.2 Fitness computation

Fitness computation is the core component in GANN model. We employ a 3-layered feedforward ANN to calculate the fitness value of each chromosome in the population. The fitness function of our model is defined as the number of correctly labelled instances returned by an ANN among input samples. The equation is presented as follow:

$$\text{fitness} f = \sum_{i=1}^{n} \sum_{k=1}^{c} s_{ik}$$

$$s_{ik} = t_{ik} - \sqrt{(A_{ik} - C_k)^2} \begin{cases} \geq f(x), & O_{ik} = T_{ik} \\ < f(x), & O_{ik} \neq T_{ik} \end{cases}$$

$$C_k = \frac{1}{s_k} \sum_{s \in k} A_{sk}$$

where s_{ik} is the sample i in class k, T_{ik} is the target output of sample i, *f(x)* is the activation function to be used in the ANN, A_{ik} represents the output activation value for sample i, C_k is the centroid value of class k and O_{ik} is the final output value generated by ANNs.

The centroid vector principle and the Euclidean distance are the most fundamental statistics to construct any computer algorithm. The centroid vector principle is laid on the use of mean (i.e. centroid) and standard deviations of classes to label samples. Since our model is on feature selection instead of sample classification, we exclude the use of standard deviations of classes. Meanwhile, the Euclidean distance measures how far the distance of individual samples are from each class. Depending on the proximity value, the sample is labelled to its nearest class. As Schwarzer et al. [13] commented:

> *With increasing number of hidden units we fit more and more implausible functions which move away from the true law f, and hence the misclassification probability increases.*

Large network sizes can lead to over-fitting problems as the network tries to fit the connection weight closer to the target output so that it can reduce the error rate of the network. Thus, to reduce the risk of over-fitting in the network, a standard 3-layered network architecture 10-5-O is applied, i.e. 10 input nodes, 5 hidden nodes and O output nodes that corresponding to the number of classes in the data sets. Four activation functions comprising binary sigmoid, linear, hyperbolic tangent (tanh) and threshold, will be compared in this study.

Table 2 Summary of the trial results based on various population sizes using the tanh system

Population sizes	100	200	300	400	500	600
Data set A (2-class)						
Fitness accuracy (%)	47.5	83.94	94.62	98.32	99.36	99.54
Median (#sample)	99	100	100	100	100	100
Selected features (30)	30	30	30	30	30	30
Data set B (3-class)						
Fitness accuracy (%)	4.44	20.36	32.56	35.2	28.7	17.14
Median	96	98	99	99	99	98
Selected features (30)	28	29	30	30	30	30

2.2.1 *Activation function in comparison*

Given w is the network weight, i is the input node to the hidden node j, b is the bias nodes and *f*(x) is the output activation value, the equations of the four types of activation function for the hidden node is as follows:

- The binary sigmoid function:

$$f(x_j) = \left[\frac{1}{1 + \exp(-x_j)}\right] + b_j.$$

- The linear function:

$$f(x_j) = \left[K * \sum^{i} w_{ij} x_j\right] + b_j.$$

- The tanh function:

$$f(x_j) = \left[\frac{2}{1 + \exp(-x_j)} - 1\right] + b_j.$$

- The threshold function:

$$f(x_j) = \left[\sum^{i} w_{ij} x_j\right] + b_j.$$

K is the constant value which set to 1 for a simple linear function and the threshold value θ, for the threshold function is 0; i.e. the hidden node will be activated when the activation value is greater than 0. Whilst, the activation range for the binary sigmoid and the tanh functions are [0, 1] and [−1, 1], respectively.

2.3 Chromosome evaluation

In the process of chromosome evaluation, the selection mechanism is responsible for selecting two fitter chromosomes for reproduction, i.e. mating. The crossover operator is responsible for introducing new chromosomes (offspring) to the next generation and the mutation operator is responsible for creating new information in the offspring.

The *tournament selection* with the tournament size of 2 is chosen in our model because it often yields a more diverse population [10] which could lead to deeper exploitation of the chromosome search and to prevent premature convergence of homogenous chromosomes. A *single-point crossover* operator with crossover rate p_c of 0.5 is applied as it is least destructive to the relationship of the units in chromosomes than the uniform crossover. A small mutation rate p_m of 0.1 is used to refresh the population with a new value into a small fraction of the chromosome.

To ensure that the quality of the chromosomes will not be distorted in the evaluation process, we apply an *elitism scheme*, in which only one chromosome is allowed to be replaced by offspring. This is to ensure that a wider exploitation of the search space is provided when the population has almost converged. Additionally, it reduces the disruption which may be caused by the mutation operator.

2.4 Termination criteria

In order for an algorithm to stop when the desired solution is achieved, a set of stopping criteria is generally required. Care must be taken determining this criteria as it may affect the generalization capability of the model in interpreting the problem.

In our design, two termination criteria are adopted. The first criterion examines the delivery of optimal results, based on the number of GA evaluation cycles, by internally repeating the process of fitness computation for chromosomes in the population. 10 GA evaluation sizes will be compared, ranging from 5,000 to 50,000. The second criterion assesses the robustness of GANN in extracting informative features by externally iterating the selection process. We set the whole process cycle to 5,000.

3 Data sets

Six data sets were considered, comprising two synthetic, two real-world data sets (microarray data sets) and two virtual screening data sets (bioassay data sets), see Table 3. The first four data sets were used to assess the influence of different activation functions and fitness evaluations on the outcome of the GANN and the remaining two were used to assess the robustness of the optimum GANN parameters for feature selection. In order to evaluate the influence of different activation functions and fitness evaluations on the outcome of the GANN, we predefined 30 significant features in each synthetic data set.

Both synthetic data sets were created from a standard normal distribution N(0,1) with the exception of the 30 predefined features.

3.1 Synthetic data set 1

The synthetic data set 1 is a 2-class data set containing 100 samples equally distributed in each class. For the synthetic data set 1, each sample is associated with 10,000 features. The 30 predefined features were from a normal distribution with $\mu = 2$ and $\sigma = 1$. This data set is designed to simulate the interactivity pattern of ALL/AML data.

3.2 Synthetic data set 2

The synthetic data set 2 contains 67 samples distributed into three distinct classes, with 20 samples for class 1, 30 samples for class 2 and the remaining 17 samples for class 3. This data set is designed to simulate the complex feature interactions in the multiclass scenario containing a high dimension of irrelevant information and inequality distribution of sample patterns available for each class. For the synthetic data set 2, each sample contained 5,000 features. Among the 30 predefined features in the synthetic data set 2, the first 10 were normally distributed with $\mu = 0.5$ and the remaining 20 features with $\mu = 2$. Both with $\sigma = 1$. Similar to the SRBCTs data set, this data set has a complex level of feature relationships and there are no clearly distinct clusters among these classes.

4 Results

The overall GANN system performance result based on four different activation functions in three population

Table 3 Summary of the data sets

Data sets	Authors	Instances	Features	Classes	Predefined/Reported sig. features	Data pre-processing
Synthetic 1	–	100	10,000	2	30	No
Synthetic 2	–	67	5,000	3	30	No
ALL/AML	Golub et al. [6]	72	7,129	2	50	No
SRBCTs	Khan et al. [8]	82	2,308	4	96	No
AID362	Schierz [12]	4,279	144	2	–	No
AID688	Schierz [12]	27,189	153	2	–	No

Table 4 Summary of the results

Data sets	Population sizes	Binarysigmoid			Linear			Tanh			Threshold		
		A	B	C	A	B	C	A	B	C	A	B	C
Synthetic 1	100	28.6	53.96	7352.7	29.9	61.87	2385.7	27.6	49.11	10789.4	29.5	56.66	2960.4
	200	30	86.01	3754.6	30	91.24	1161.0	30	82.18	5844.2	30	88.31	1503.8
	300	30	93.49	3079.4	30	97.18	1040.2	30	91.92	4667.1	30	96.87	1294.0
Synthetic 2	100	5.5	5.61	9375.5	8.9	10.99	4045.9	3.8	3.95	12838.5	7.3	7.74	4429.0
	200	11.3	19.40	8673.8	15.3	34.74	3470.3	9.7	14.28	12195.6	13.3	26.87	3973.1
	300	13	29.80	8368.8	16.3	50.45	3173.9	10.8	22.80	11928.9	15.3	41.05	3756.3
ALL/AML	100	47.4	77.83	11081.3	54.1	67.76	1658.9	47.4	77.95	1790.6	47.8	59.84	2251.7
	200	48.1	88.41	8318.7	55.9	75.98	1523.9	47.7	88.33	1427.8	45.6	65.57	2215.7
	300	42.5	90.68	7919.4	50.8	79.05	1573.2	42.1	90.86	1468.1	41.4	66.79	2491.9
SRBCTs	100	59.3	59.42	6402.8	51.1	46.53	3298.4	53.4	53.85	9362.3	55.3	52.13	3335.6
	200	56.6	73.31	5378.3	55.7	60.10	3039.6	51.2	67.69	8455.5	58.2	66.43	2922.5
	300	48.1	75.27	6436.8	50.6	63.79	3509.4	44.1	70.18	9379.7	50.3	69.52	3536.2

sizes is tabulated in Table 4, where ***A*** is the average number of significant features selected with a minimum of 50 frequency selections; ***B*** is the average fitness performance in percentage achieved for computing fitness value and ***C*** is the processing time in seconds. Results for individual systems performance based on different evaluations sizes are presented in Figs. 1, 2, 3 and Table 5.

We analysed the results with respect to the number of significant features selected based on a minimum of 50 frequency selections, the fitness accuracy of different systems and the processing time needed by each system. We did not consider the use of classification in this study, but instead of using two synthetic data sets, each with a predefined number of significant features; to assess the efficiency of different GANN systems. The ability to extract features from the microarray data sets will be compared to the performance of the synthetic problems.

It is important to appreciate that the GANN algorithm does not consider the semantic content of the data. Except insofar as it knows that there exist classes with the data. So any manipulation of the data before submission to the algorithm could affect the outcome. Moreover, the success of classification is dependent on several related factors, including data structures (number of positive and negative instances, number of classes, feature correlation, feature distribution), data pre-processing (filtering, normalisation, imputation, selection), validation procedures (training and test set distribution, leave-one-out cross-validation, K-fold cross-validation) and the number of repetitions for a classification method.

Results show significant improvement on all four systems with increased population sizes. Binary class data sets, i.e. synthetic 1 and ALL/AML; have better system performance comparing to multiclass data sets, i.e. synthetic 2 and SRBCTs.

The tanh system has a very low processing time in every population in the ALL/AML data set compared to the linear and threshold based systems. Overall, the tanh system outperforms the other three systems in the ALL/AML data set. Two main reasons for its efficiency are:

(a) The data set contains subgroup of cancer classes within a known class and has a large value interval within a gene in the data sets.

(b) The bipolar range [−1,1] in the tanh system has produced two output signals, i.e. positive and negative, in the output of the network, which has expanded the differentiation between the classes in the data set. This has reduced the chances of mislabelling the instance into the wrong class.

Results also show that the performance of each system is highly dependent on the quality of the data set and, to some extent, the population size and the degree of statistics involved in the fitness computation process. Depending on the requirement of the study, each system has its pros and cons in terms of the quality of the selected features within an acceptable confidence range and in an acceptable processing time. We observed the following:

1. The linear system has the lowest processing time in all data sets. However, the identified features did not acceptably classify the data.
2. The sigmoid and tanh systems provide a high fitness confidence in larger population sizes but are computationally cost intensive.

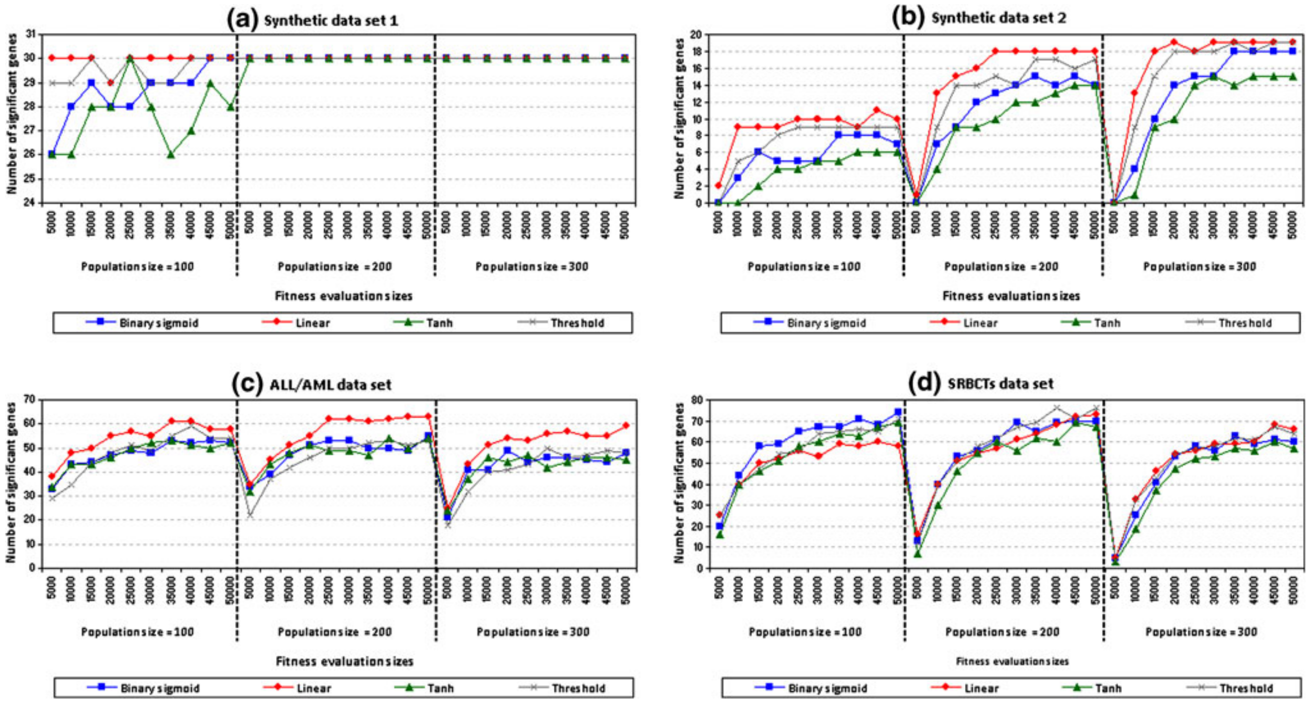


Fig. 1 The number of features selected by each system based on the selection frequency of 50 and above in various sizes of population and fitness evaluation

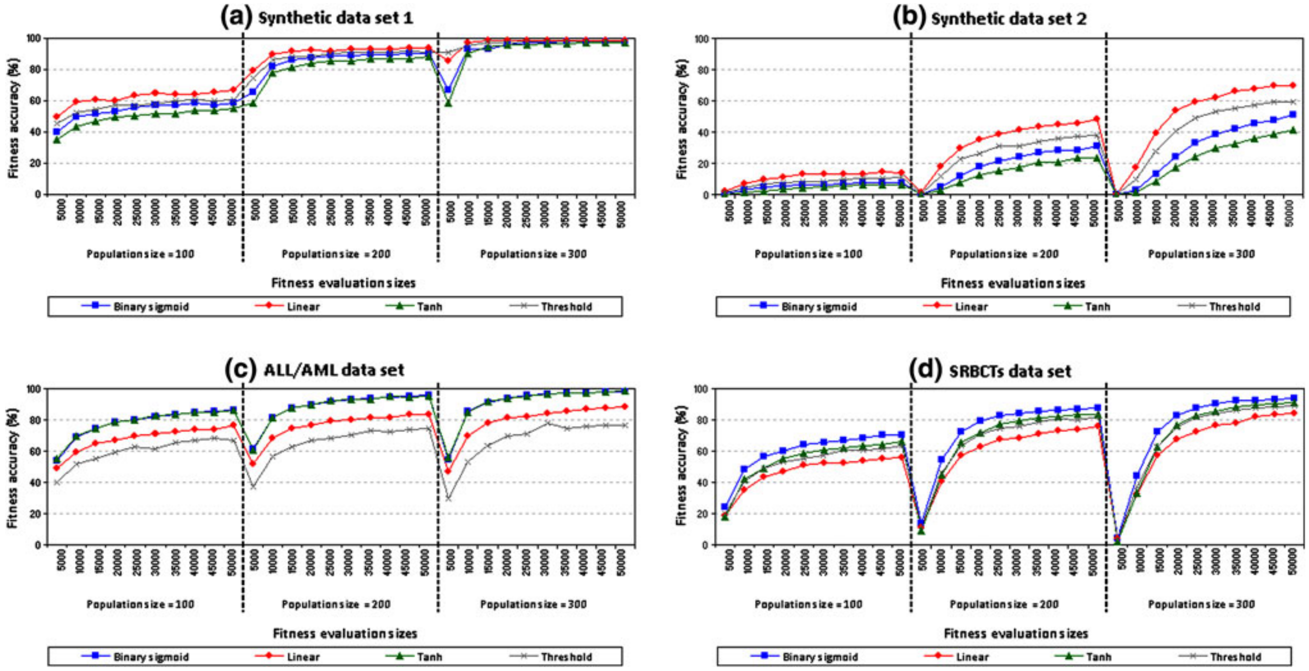


Fig. 2 The fitness performance by each system in various sizes of population and fitness evaluation

3. Conversely, the threshold system is unable to effectively model data with multiple subclasses in a known class.

In the next section, we further investigate the relation between GA population, GA fitness evaluation and ANN activation function in extracting significant features.

4.1 The number of significant features

From Tables 4–5 and Fig. 1, results show a direct relation between population sizes, evaluation sizes and number of selected features. We observe the following:

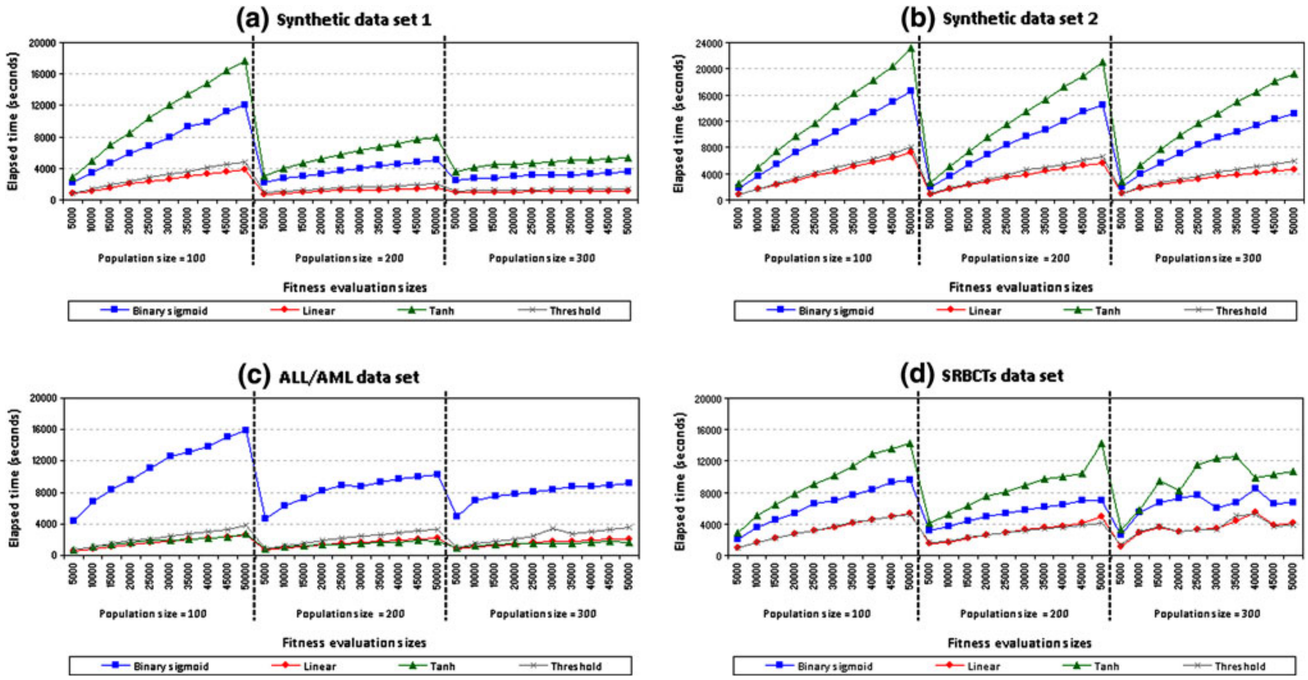


Fig. 3 The processing time of each system in various sizes of population and fitness evaluation

1. For synthetic data sets in Fig. 1a, b, none of the systems able to detect all 30 predefined features in population size 100. When the population size was increased, all systems have found all predefined features in synthetic data set 1 and a significant increased number of predefined features with the increased number of fitness evaluations in synthetic data set 2.
2. For microarray data sets in Fig. 1c, d:

(a) There were a significantly increased number of identified features by all systems when the fitness evaluation size was increased in a larger population size.
(b) An increased number of overlapping features found by all systems in Table 4 with the increased sizes of fitness evaluation and population.
(c) There were a highest number of selected features by all systems in the population size 100 than the features selected in the population sizes 200 and 300. These selected features were mostly overlapped with the features being reported by the corresponding studies.
(d) All systems have a lowest number of selected features in the population size 300. A marginal decrease on the overlapping features identified when compared to the features selected in population sizes 100 and 200.
(e) As the population size increased, all systems pruned unnecessary features from the search process.

3. In every case there were fewer features identified by all systems in the fitness evaluation size 5,000 for every population. When the fitness evaluation size was increased, more significant features were found.
4. In all cases the linear system had the highest number of selected features and the tanh system the lowest.
5. Overall, the linear and the tanh systems have the best performance when the fitness evaluation size exceeds 20,000 in every population size. The sigmoid system required at least 30,000 fitness evaluations in every population size for consistent results.
6. There is a discrepancy in the minimum fitness evaluation size for different types of data sets in the threshold system. For binary data sets (synthetic data set 1 and ALL/AML), a minimal of 25,000 fitness evaluations is required for better results. For multiclass data sets (synthetic data set 2 and SRBCTs) the threshold system required at least 35,000 fitness evaluations for satisfactory results.

This observation verified that a small population size, i.e. 100, is not efficient for high dimensional data, such as microarray data. This may be due to limited space in the population needed to accommodate the possible combination of features within the data. Larger populations, i.e. 200 and 300, are recommended for high dimensional data sets.

4.2 The fitness performance

Results show a significant improvement in the fitness performance of each GANN system with the increased

Table 5 The selected predefined features on the synthetic data sets and the overlapped features on the microarray data sets to its original studies

Data sets	Fitness evaluations	Binarysigmoid			Linear			Tanh			Threshold		
		P100	P200	P300	P100	P200	P300	P100	P200	P300	P100	P200	P300
Synthetic 1	5,000	26	30	30	30	30	30	25	30	30	29	30	30
	10,000	28	30	30	30	30	30	26	30	30	29	30	30
	15,000	29	30	30	30	30	30	28	30	30	30	30	30
	20,000	28	30	30	29	30	30	28	30	30	29	30	30
	25,000	28	30	30	30	30	30	28	30	30	29	30	30
	30,000	29	30	30	30	30	30	28	30	30	29	30	30
	35,000	29	30	30	30	30	30	26	30	30	30	30	30
	40,000	29	30	30	30	30	30	27	30	30	30	30	30
	45,000	30	30	30	30	30	30	29	30	30	30	30	30
	50,000	30	30	30	30	30	30	28	30	30	30	30	30
Synthetic 2	5,000	0	0	0	2	1	0	0	0	0	0	0	0
	10,000	3	7	4	9	13	13	0	4	1	5	9	9
	15,000	6	9	10	9	15	18	2	9	9	6	14	15
	20,000	5	12	14	9	16	19	4	9	10	8	14	18
	25,000	5	13	15	10	18	18	4	10	14	9	15	18
	30,000	5	14	15	10	18	19	5	12	15	9	14	18
	35,000	8	15	18	10	18	19	5	12	14	9	17	20
	40,000	8	14	18	9	18	19	6	13	15	9	17	18
	45,000	8	15	18	11	18	19	6	14	15	9	16	19
	50,000	7	14	18	10	18	19	6	14	15	9	18	19
ALL/AML	5,000	14	17	10	19	19	12	17	15	11	11	10	9
	10,000	18	19	20	21	21	21	18	20	17	16	18	16
	15,000	20	19	17	21	22	22	19	20	20	19	21	20
	20,000	21	21	20	22	23	22	20	19	19	21	21	17
	25,000	21	20	18	23	23	22	20	20	21	21	21	21
	30,000	20	20	19	23	23	23	23	20	18	21	21	22
	35,000	22	21	19	22	24	23	24	20	19	22	23	21
	40,000	22	20	19	23	25	23	21	21	21	22	22	22
	45,000	23	20	17	22	23	22	22	21	20	22	23	21
	50,000	22	21	20	22	24	22	22	22	19	21	23	21
SRBCTs	5,000	19	13	5	22	14	4	16	7	3	22	11	4
	10,000	32	33	22	28	28	25	33	25	17	26	29	26
	15,000	29	27	33	32	31	29	37	38	32	31	34	30
	20,000	40	36	38	32	33	33	36	40	37	37	35	35
	25,000	44	39	38	34	34	34	41	43	38	36	38	35
	30,000	46	44	37	33	36	34	44	41	39	37	41	35
	35,000	44	41	39	36	37	37	45	42	43	40	39	37
	40,000	48	46	39	36	40	34	43	43	42	42	43	36
	45,000	45	43	38	35	39	37	47	46	43	40	40	38
	50,000	47	45	39	35	38	35	49	45	43	43	41	36

sizes of fitness evaluation and population, as is indicated in Fig. 2 and Table 4.

For synthetic data sets in Fig. 2a, b, increasing the population size from 100 to 200 improved the fitness accuracy based on the selected features by each system in every fitness evaluation. The performance of each system also improved in the synthetic data set 2 when the population size was increased to 300. With the comparison of the fitness performance of each system based on two population sizes: 200 and 300, in the synthetic data set 1, there is no significant performance difference in every fitness evaluation in each system, except that the threshold system had the best performance in the fitness evaluation size 5,000 in the population size 300.

From Fig. 2 and Table 4 there seems to be a clear relation between data quality and the fitness performance of each system. An important discrepancy in the effect of the stronger features (i.e. features with significant differential characteristics) can severely affect the fitness performance of the system. This discrepancy could subsequently affect the final decision-making. We observe the following:

1. For synthetic data set 1, all four systems have identified all 30 predefined features in every fitness evaluation in the population sizes 200 and 300. However, a slightly better fitness performance was achieved by each system in population size 300 than in population size 200. Among these, the linear system marginally outperformed the others.
2. For synthetic data set 2, the linear and the threshold systems have a significant difference in the fitness performance in the population size 300, although, both systems have a minor different on the number of predefined features being identified. This fitness discrepancy may be due to the influence of some stronger (fitter) predefined features which resulted in a better fitness performance. This inconsistency may be also due to the selection of some strong noisy features in the linear system. This indicates that the linear system could explore stronger features more efficiently than the other systems. However, it cannot be assumed that the identified features are features of interest.
3. For ALL/AML data set, the sigmoid and the tanh systems have a better overall fitness performance than the linear system, although more features, as well as more overlapping features, have been identified by the latter system, as is indicated in Fig. 1 and Table 4. The threshold system has the lowest fitness performance in every population size and this may be due to the involvement of multiple subclasses within a known class in ALL/AML data set.
4. For SRBCTs data set, the linear system has the lowest fitness performance, while the sigmoid system outperformed the other others. There is a marginal performance difference between tanh and threshold systems.
5. Across the board, the sigmoid, linear and tanh systems have a lower fitness performance in the SRBCTs data set than in the ALL/AML data set. Conversely, the threshold system has a slightly better performance in the SRBCTs data set than in the ALL/AML data set. This may be due to the tumour classes in the SRBCTs data set are not directly related to each other, while the ALL/AML data set is formed by subtypes of similar cancer class.

Results also show a clear relation between fitness evaluation sizes, population sizes and fitness performance achieved by each system. In microarray cases, there is a low fitness performance achieved by each system in every fitness evaluation in the population size 100. With the increased population size to 300, the performance of each system was significantly improved. We observe the following:

1. For ALL/AML data set, both the sigmoid and tanh systems have the best fitness performance with minimally 30,000 fitness evaluations in the population size 200 and above, while the linear and the threshold based systems required only 20,000 fitness evaluations.
2. For SRBCTs data set, all systems have a better performance in the population size 300 than in the population size 200, with minimally 30,000 fitness evaluations. This may be due to the multiclass nature of the data set.

4.3 The processing time

From Fig. 3, there is no clear relation between population sizes, fitness evaluation sizes and the processing time.

A high ratio of elapsed time in different fitness evaluation sizes by each system was found for a given population size. There appears to be no significant processing time difference between similar sets of fitness evaluations in population sizes 200 and 300. This suggests that the processing time is associated with a high fitness evaluation size rather than the population size. However, in the population size 100, a higher processing time is required by each system in every fitness evaluation when compared to the identical sets of fitness evaluations in a larger population size. We conjecture that this is because a small population size lacks the capability to accommodate more learning patterns (chromosomes) for the system to model the general rules. The reduced capacity lessens the fitness performance of the systems and increases processing time, even though a sufficient number of evaluations are allowed.

Results show a clear relation between activation functions and the processing time required. We observe that:

1. Both the linear and threshold systems in all cases have the lowest processing time. The sigmoid and tanh systems have the highest computational cost.
2. For ALL/AML data set, there is no significant difference on the processing time between linear and tanh systems. This may be due to the ALL/AML data set containing multiple subclasses of instances within a

known class and the tanh function may be more adapted to nonlinear problems.

3. There is no strong relation between the number of classes, the dimensionality of the data and the time needed by each system to compute the fitness values of each chromosome in the population.

4.4 The bioassay data sets

In the previous sections, we discussed the GANN performance to handle data with high feature dimension, sample scarcity and complex feature interaction. Based on these findings, we observed that the tanh function is, among all the systems, the most effective to extract the most significant features from the data sets.

In this section, we examine the selection performance of the tanh system to handle bioassay data characterized by low feature dimension, feature-independent and highly imbalanced between the number of positive and negative compounds (i.e. samples). The experiments based on the fitness evaluation size 30,000 within population size 300 and ANN structure's 20-10-2 were applied. The completeness of the findings is compared with the original work reported by Schierz [12] and the principal component analysis (PCA). The data sets have been split into an 80% training set and a 20% test set and a tenfold cross-validation procedure, as recommended in the original work and evaluated using four cost-sensitive classifiers (CSC), i.e. Naïve Bayes (NB), Random Forest (RF), Support Vector Machine (SMO) and Classification Tree (J48), constructed in the WEKA environment. The identical WEKA parameters as indicated in the original work were used. The top 20% of the attributes from each data set were selected by GANN system and the summary results are presented in Figs. 4 and 5.

Results show a significantly improved performance using the GANN system when compared to PCA and the original work. Results indicate that:

1. For the AID362 data set:

(a) Using the 38 attributes selected by GANN, a CSC RF has produced better results than using the entire 144 attributes reported by Schierz [12] and the 95 attributes selected by PCA.

(b) A significant decrease on the number of false positive (FP) rate in all classifiers, except a CSC SMO, based on the attributes selected by GANN than the attributes selected by PCA.

2. For the AID688 data set:

(a) Using the 31 attributes selected by GANN, a MetaCost J48 tree has performed better than using the entire 153 attributes reported by Schierz [12] and the 116 attributes selected by PCA.

(b) The CSC RF which was unable to run using neither the whole data set nor the 116 attributes selected by PCA, due to the high sample size, has produced good results using the attributes selected by GANN.

(c) A decrease in performance using CSC NB and CSC SMO for attributes selected by GANN when compared to using the entire attributes and the attributes selected by PCA.

(d) A significant improved FP rate in all classifiers using the attributes selected by GANN.

The findings demonstrate a comparably better performance using the GANN prototype than PCA and Schierz's work. The benefit of using GANN is that it enables computationally effective algorithms, such as Random Forest and Classification Tree, to be implemented with a large data set and with a high success rate. Considering the

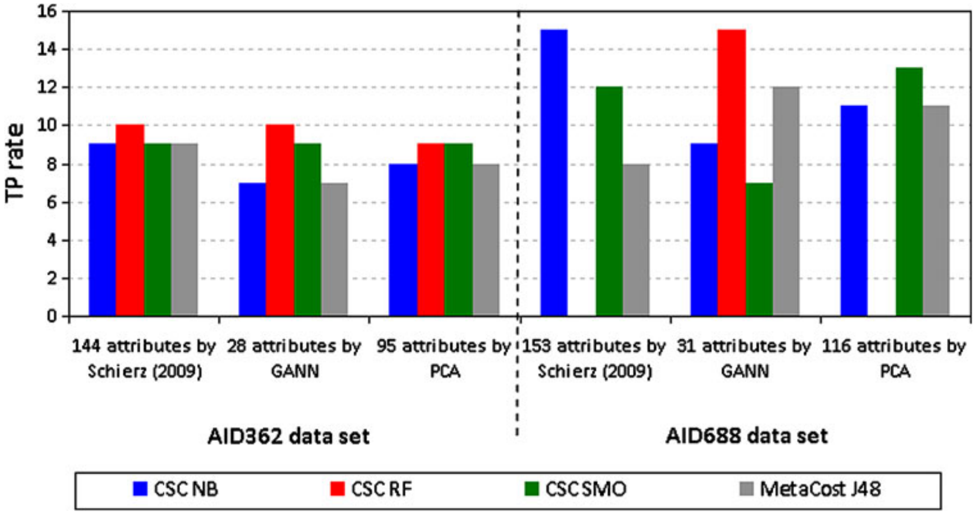
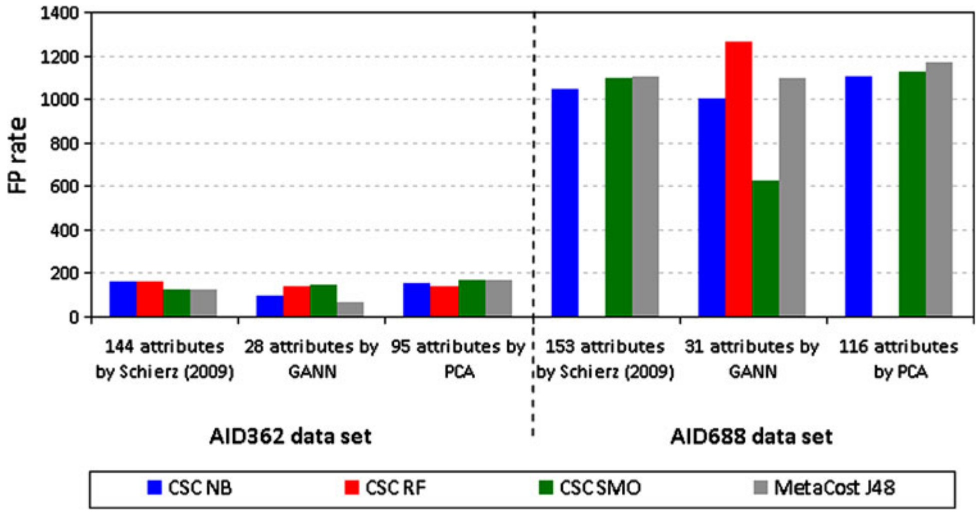


Fig. 4 The true positive (TP) rate of the bioassay data sets with under or approximately a 20% false positive (FP) rate

Fig. 5 The false positive (FP) rate of the bioassay data sets



GANN only implemented 20% of the attributes from the data set, the classification performance of cost-sensitive classifiers has not been compromised. This shows that the GANN system has successfully identified the most significant attributes needed to discriminate compound classes in the data sets. However, the only downside of the GANN system is that a computationally intensive processing time required for bioassay data sets than for microarray data sets.

5 Discussion and conclusions

The results indicate that the capability of the hybrid GA/ANN system in finding significant discrimination features is dependent upon the configuration of the GA population, the GA fitness evaluation and the selection of ANN activation functions.

Cartwright [3] commented that the population size is not critical in the success of a GA, provided that the population size is not unreasonably small (i.e. <40 chromosomes). However, our results indicate the importance of the population size in the success of a GA and a strong interaction on the evolution process, as is observed by DeJong and Spears [5]. This is indicated in the population size 100 with the elevated processing time and low fitness accuracy achieved by each system in every fitness evaluation. With the increased population size, better fitness performance and lower processing time are achieved when similar fitness evaluations were applied. DeJong and Spears [5] made this observation based on the augmentation in the crossover operator and we derive similar conclusions with the increased fitness evaluation sizes.

In addition to the population size, a larger fitness evaluation also promise better fitness confidence in the selected features and the processing time is not always increased with larger evaluations. This is indicated in Fig. 3, where a lower or equal ratio of elapsed time was found in each fitness evaluation in population sizes 200 and 300.

Despite the ideal population size, i.e. ranging from 40 to 100, as suggested by Cartwright [3], our results show that the population size 100 is still too small for handling high dimensional data sets, microarray data specifically. Based on the ALL/AML and SRBCTs data sets, our findings suggest the minimal population size for microarray data should be between 200 and 300. The minimal fitness evaluation size for binary class data should be 20,000 and 25,000 for multiclass data. The maximal fitness evaluation size should not exceed 40,000.

Our findings suggest that the linear and the tanh systems are the two most effective ANN activation functions to be used to compute fitness values for GA chromosomes.

The summary of our findings:

1. The linear system is able to explore the potential features more effectively than the other three and is less computationally demanding and is consequently more appropriate for high repetition statistics. On the other hand this system seems to find a high number of low interest features.
2. The sigmoid system is also able to efficiently explore the potential feature space. However, this system requires a more intensive processing cost than the other three.
3. The threshold system lacks of the stability in extracting consistent features and appears sensitive to data distribution.
4. The tanh system appears to be the most effective for finding the most significant features within the data sets.
5. All systems have a significant improvement on their performance with the population sizes 200 and 300 and

the fitness evaluation sizes, ranging from 2,000 to 40,000.

6. The tanh based GANN system has produced better results in large and imbalanced bioassay data sets. However, it is computational intensive.

References

1. Beiko RG, Charlebois RL (2005) GANN: genetic algorithm neural networks for the detection of conserved combinations of features in DNA. BMC Bioinformatics 6:36
2. Bevilacqua V, Mastronardi G, Menolascina F, Paradiso A, Tommasi S (2006) Genetic algorithms and artificial neural networks in microarray data analysis: a distributed approach. Eng Lett Spec Issue Bioinformatics 13(3):335–343
3. Cartwright H (2008) Using artificial intelligence in chemistry and biology: A practical guide. In: Chapter Evolutionary Algorithms, CRC Press, Taylor & Francis Group, Boca Raton, London pp 113–172
4. Cho HS, Kim TS, Wee JW, Jeon SM, Lee CH (2003) cDNA microarray data based classification of cancers using neural networks and genetic algorithms. In Nanotech'03: Nanotechnology Conference and Trade Show, proceedings, vol 1
5. DeJong KA, Spears WM (1991) An analysis of the interacting roles of population size and crossover in genetic algorithms. In: Schwefel HP, Männer R (eds) PPSN'91: first Workshop on Parallel Problem Solving from Nature, proceedings, volume 496 of Lecture Notes in Computer Science, Springer, Berlin pp 38–47
6. Golub TR, Slonim DK, Tamayo P, Huard C, Gaasenbeek M, Mesirov JP, Coller H, Loh ML, Downing JR, Caligiuri MA, BloomfIeld CD, Lander ES (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286(5439):531–536
7. Karzynski M, Mateos Á, Herrero J, Dopazo J (2003) Using a genetic algorithm and a perceptron for feature selection and supervised class learning in DNA microarray data. Artif Intell Rev 20(1–2):39–51
8. Khan J, Wei JS, Ringnér M, Saal LH, Ladanyi M, Westermann F, Berthold F, Schwab M, Antonescu CR, Peterson C, Meltzer PS (2001) Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nat Med 7:673–679
9. Lin T-C, Liu R-S, Chao Y-T, Chen S-Y (2006) Multiclass microarray data classification using GA/ANN method. In Yang Q, Webb GI (eds) PRICAI'06: trends in artificial intelligence, ninth Pacific Rim international conference on artificial intelligence, proceedings, vol 4099 of Lecture Notes in Computer Science. Springer, pp 1037-1041
10. Mitchell TM (1997) Does machine learning really work? AI Mag 18(3):11–20
11. Ramasubramanian P, Kannan A (2006) A genetic-algorithm based neural network short-term forecasting framework for database intrusion prediction system. Soft Comput 10:699–714
12. Schierz A (2009) Virtual screening of bioassay data. J Cheminform 1:2
13. Schwarzer G, Vach W, Schumacher M (2000) On the misuses of artificial neural network for prognostic and diagnostic classification in oncology. Stat Med 19(4):541–561
14. Shenouda E (2006) A quantitative comparison of different MLP activation functions in classification. In: ISNN'06: advances in neural networks, third international symposium on neural networks, proceedings, Part I–III, vol 2971 of Lecture Notes in Computer Science. Springer, Berlin, pp 849–857
15. Taheri M, Mohebbi A (2008) Design of artificial neural networks using a genetic algorithm to predict collection efficiency in venturi scrubbers. J Hazard Mater 157(1):122–129
16. Verma B, Zhang P (2007) A novel neural-genetic algorithm to find the most significant combination of features in digital mammograms. Appl Soft Comput 7(2):612–625
17. Zorić G, Pandžić IS (2006) Real-time language independent lip synchronization method using a genetic algorithm. Signal Processing 86(12):3644–3656