We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Feature Selection for Classification with Artificial Bee Colony Programming

Sibel Arslan and Celal Ozturk

Abstract

Feature selection and classification are the most applied machine learning processes. In the feature selection, it is aimed to find useful properties containing class information by eliminating noisy and unnecessary features in the data sets and facilitating the classifiers. Classification is used to distribute data among the various classes defined on the resulting feature set. In this chapter, artificial bee colony programming (ABCP) is proposed and applied to feature selection for classification problems on four different data sets. The best models are obtained by using the sensitivity fitness function defined according to the total number of classes in the data sets and are compared with the models obtained by genetic programming (GP). The results of the experiments show that the proposed technique is accurate and efficient when compared with GP in terms of critical features selection and classification accuracy on well-known benchmark problems.

Keywords: feature selection, classification algorithms, evolutionary computation, genetic programming, artificial bee colony programming

1. Introduction

In recent years, data learning and feature selection has become increasingly popular in machine learning researches. Feature selection is used to eliminate noisy and unnecessary features in collected data that can be expressed more reliably and high success rates are obtained in classification problems. There are several works which related to solve genetic programming (GP) in feature selected classification problem [1–4]. Since artificial bee colony programming (ABCP) is a recently proposed method, there is no work related to this field. In this chapter, we evaluated the success of classification by selecting the features of GP and ABCP automatic programming methods using different data sets.

1.1 Goals

The goal of this chapter is classify models are obtained with comparable accuracy to alternative automatic programming methods. The overall goals of chapter are set out below.

1. Evaluation of the performance of models with parameters such as classification accuracy, complexity.

2. Whether ABCP method actually can select related/linked features.

3. Evaluating training performance of automatic programming methods to determine if there is overfitting.

The organization of the chapter is as follows: background is described in Section 2, detailed description of GP and ABCP is introduced in Section 3. Then, experiments and results are presented and discussed in Section 4. The chapter is concluded in Section 5 with summarizing the observations and remarking the future work.

2. Background

2.1 Feature selection

Feature selection makes it possible to obtain more accurate results by removing irrelevant and disconnected features in model prediction. The model prediction provides the functional relationship between the output parameter y and the input parameters x of the data set. Removing irrelevant features reduces the dimension of the model, thus it reduces space complexity and computation time [5, 6].

Feature selection methods are examined in three main categories as filter methods, embedded methods and wrapper methods [7, 8]. Filtering methods evaluate features with the selection criterion based on correlations between features (feature relevance) and redundancy and associate of features with class label vectors. Wrapper methods take into account the success of classification accuracy and decide whether or not an object will be included in the model. In order to obtain the successful model, it is not preferred in time constrained problems because the data set is trained and tested many times [9]. Embedded methods perform feature selection as part of model construction is based on identifying the best divisor.

In recent years, increasing interest in discovering potentially useful information has led to feature selection researches [10–15]. In [10], a spam detection method of binary PSO with mutation operator (MBPSO) was proposed to reduce the spam labeling error rate of non-spam email. The method performed more successful than many other heuristic methods such as genetic algorithm (GA), particle swarm optimization (PSO), binary particle swarm optimization (BPSO), and ant colony optimization (ACO). Sikora and Piramuthu suggested GA for feature selection problem using Hausdorff distance measure [11]. GA was quite successful the accuracy of prediction accuracy and computational efficiency in real data mining problems. In [12], a wrapper framework was proposed to find out the number of clusters in conjunction in the selection of features for uncontrolled learning and normalize the tendencies of feature selection criteria according size. Feature subset selection using expectation maximization clustering (FSSEM) was used as the performance criterion for the maximum likelihood. Schiezaro and Pedrini proposed a feature selection method based on artificial bee colony (ABC) [13]. The method presented better results for the majority of the data sets compared to ACO, PSO, and GA. Yu et al. showed that selecting the discriminative genes of GP and expressing the relationships between the genes as mathematical equations were proof that GP has been applied feature selector and cancer classifier [2]. Landry et al. compared k-nearest neighbor (k-NN) with decision trees generated by GP in several

benchmark datasets [14]. GP was more reliable performance for feature selection and classification problems. Our chapter is the first to work with the ABCP's ability to select the necessary features in datasets.

2.2 Classification

Classification provides a number of benefits to make it easier to learn about data and to monitor the data. Several researches have been applied to solve the classification problems [15–17]. Fidelis et al. classified each chromosome based on GA that represented classification rules [15]. The algorithm was evaluated in different data sets and achieved successful results. A new algorithm was proposed to learn the distance measure for the closest neighbor classifier for k-nearest multi class classification in [16]. Venkatesan et al. proposed progressive technique for multi class classification can learn new classes dynamically during the run [17].

Much work has been devoted to classification using GP and ABC [18–25]. GP based feature selection age layered population structure as a new algorithm for feature selection with classification was compared with other GP versions in [18]. Lin et al. proposed the feature layered genetic programming method for feature selection and feature extraction [19]. The method, had a multilayered architecture, was built using multi population genetic programming. The experimental results show that the method achieved high success in both feature selection and feature extraction as well as classification accuracy. Ahmed et al. aimed at automatic feature selection and classification of mass spectrometry data with very high specificity and small sample representation using GP [20]. GP achieved higher success as a classification method by selecting fewer features than other conventional methods. Liu et al. designed a new GP based ensemble system to classify different cancer types where the system was used to increase the diversity of each ensemble system [21]. ABC was used data clustering on benchmark problems and was compared conventional classification techniques in [22]. Karaboga et al. applied ABC on training feed forward neural networks and classified different datasets [23]. ABC was used to improve the performance of classification in several domains avoiding the issues related to band correlation in [24]. Chung et al. proposed ABC as a new tool for data mining particularly in classification and compared evolutionary techniques, standard algorithms such as naive Bayes, classification tree and nearest neighbor (k-NN) [25]. Works showed that GP and ABC are successful in classification area. In this chapter is the first work to compare GP and recently proposed ABCP method in feature selected classification.

3. GP and ABCP

This section explicitly details GP and ABCP automatic programming methods.

3.1 GP

GP, most well-known method, was developed by Koza [26]. GP has been applied to solve numerous interesting problems [27–29]. The basic steps for the GP algorithm are similar to the steps of genetic algorithm (GA) and use the same analogy as GA. The most important difference GP and GA is representation of individuals. While GA express individuals as fixed code sequences, GP express them as parse trees. Flow chart of GP is given in **Figure 1** [30].

The first step in the flow chart is the creation of the initial population. Each individual in the population is represented by a tree where each component is called
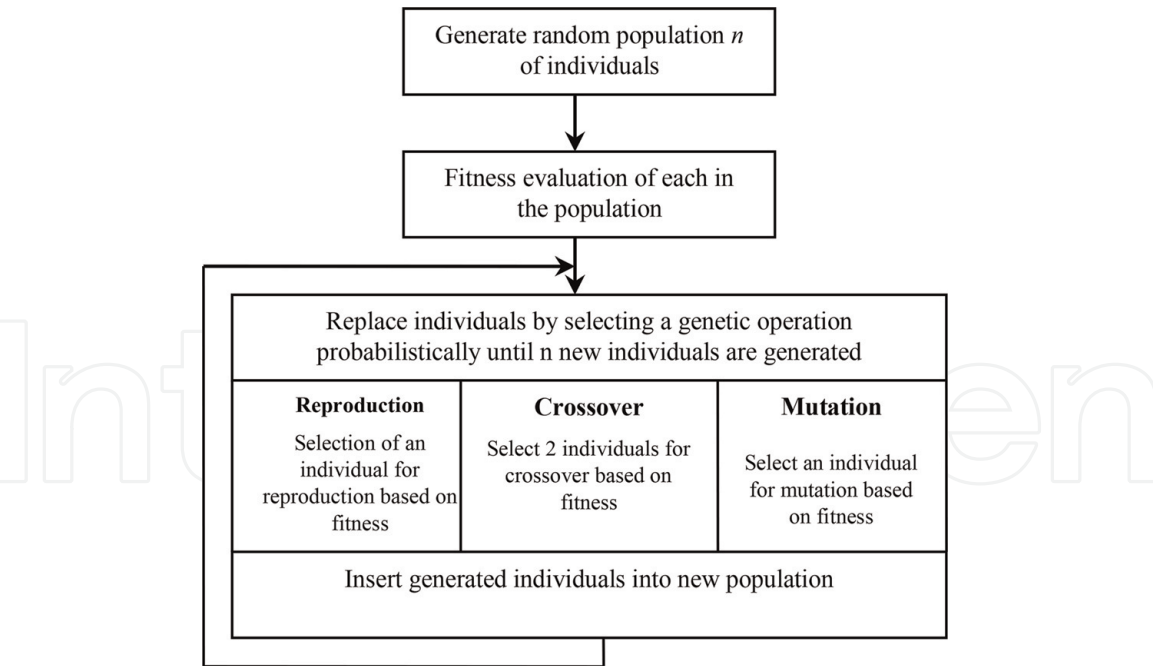
Figure 1.
The flow chart of GP.

node. The production of tree nodes is provided by terminals (constants or variables such as x, y, 5) and functions (arithmetic operators such as +, −/, sin, cos). Individuals are produced by the full method, the grow method, or the ramped half and half method [31]. Individuals are evaluated predetermined objective function. GP aims to increase the number of individuals with high quality survival and to decrease the number of low quality individuals. Individuals with high quality are more likely to pass on to the next generation. Individuals are developed them with exchange operators such as reproduction, crossover and mutation. Choosing the best individuals according to fitness are applied with methods like tournament, roulette wheel [32]. The crossover operator allows hybrid of two selected individuals to produce a new individual. Generally, sub-trees taken from two crossing points selected from parent trees are crossed to obtain new hybrid trees. The mutation operator provides unprecedented and unexplored individual elements [33]. Substitution of randomly selected tree instead of randomly selected node in the tree is called subtree mutation. Another method of mutation is a single point mutation. In this method, if the terminal is selected randomly from the tree, it is changed with the value selected from the terminal set. If the function is selected randomly from the tree, the value is selected from the function set. The best individuals of the previous generation are transferred to the current generation with elitism operator. The program is terminated when it is reached according to predefined stopping criteria such as the specific fitness value of the individuals, the number of generations.

3.2 ABCP

ABC algorithm was developed by Karaboga, modeling the food source search the intelligent foraging behavior of a honey bee swarm [34]. ABCP that was inspired ABC was introduced first time as a new method on symbolic regression [35]. In ABC, the positions of the food sources, i.e., solutions, are carried out with fixed size arrays and displays the values found by the algorithm for the predetermined variables as in GA. In the ABCP method, the positions of food sources are expressed in tree structure that is composed of different combinations of terminals and functions

that are specifically defined for problems. The mathematical relationship of the solution model in ABCP can be represented the individuals in **Figure 2** is described Eq. (1). In these notations, x is used to represent the independent, and $f(x)$ is dependent variable.

$$f(x) = 3.75\pi x - (\log 5 - \sin (2y)) \tag{1}$$

In the ABCP model, the position of a food source is defined as a possible solution and nectar of the food source is defined for the quality of the solution. There are three different types of bees, as in the ABC: employed bee, onlooker bee and scout bee in the ABCP algorithm. Employed bees are responsible for bringing the hive of nectar from specific sources that have been previously discovered and they share information about the quality of the source with the onlooker bees. Every food source is visited by one employed bee who then takes nectar to hive. The onlooker bees monitor the employed bees in hives and turn to a new source using the information shared by the employed bees. After employed and onlooker bees complete the search processes, source are checked whether source nectars are exhausted. If a source is abandoned, the employed bee using the source becomes the scout bee and randomly searches for new sources. The main steps of ABCP algorithm is given in the flow chart of ABCP algorithm in **Figure 3**.

In ABCP, the production of solutions and the determination of the quality of solutions are carried out in a similar way to GP. In the initialization of the algorithm, solutions are produced by the full method, the grow method, or the ramped half and half method [26]. The quality of solutions is found by analyzing each tree according to fitness measurement procedure.

In employed bee phase, candidate solution is created using information sharing mechanism which is the most fundamental difference between ABC and ABCP [36]. In this mechanism, when a candidate solution (v_i) is generated, the neighbor node solution x_k, taken from the tree, is randomly selected considering the predetermined probability p_{ip}. The node selected from the neighbor solution x_k determines what information will be shared with the current solution and how much it will be shared. Then node x_i, which represents the current solution in the tree that determines how to use the neighboring node, is randomly selected in the probability distribution of p_{ip}. The candidate solution v_i is produced by replacing
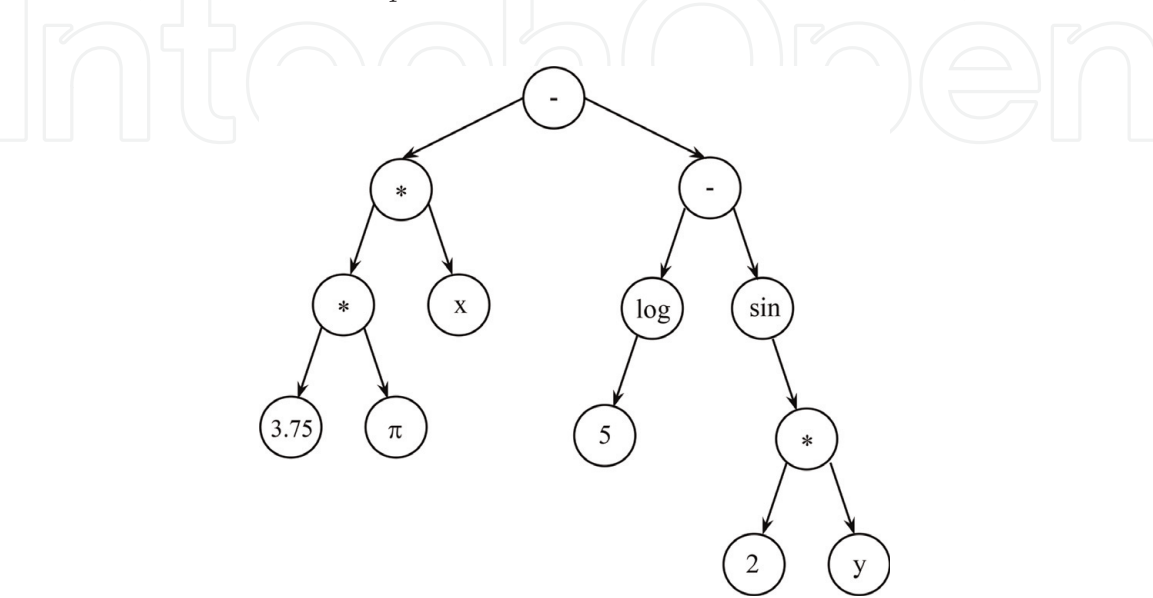

Figure 2.
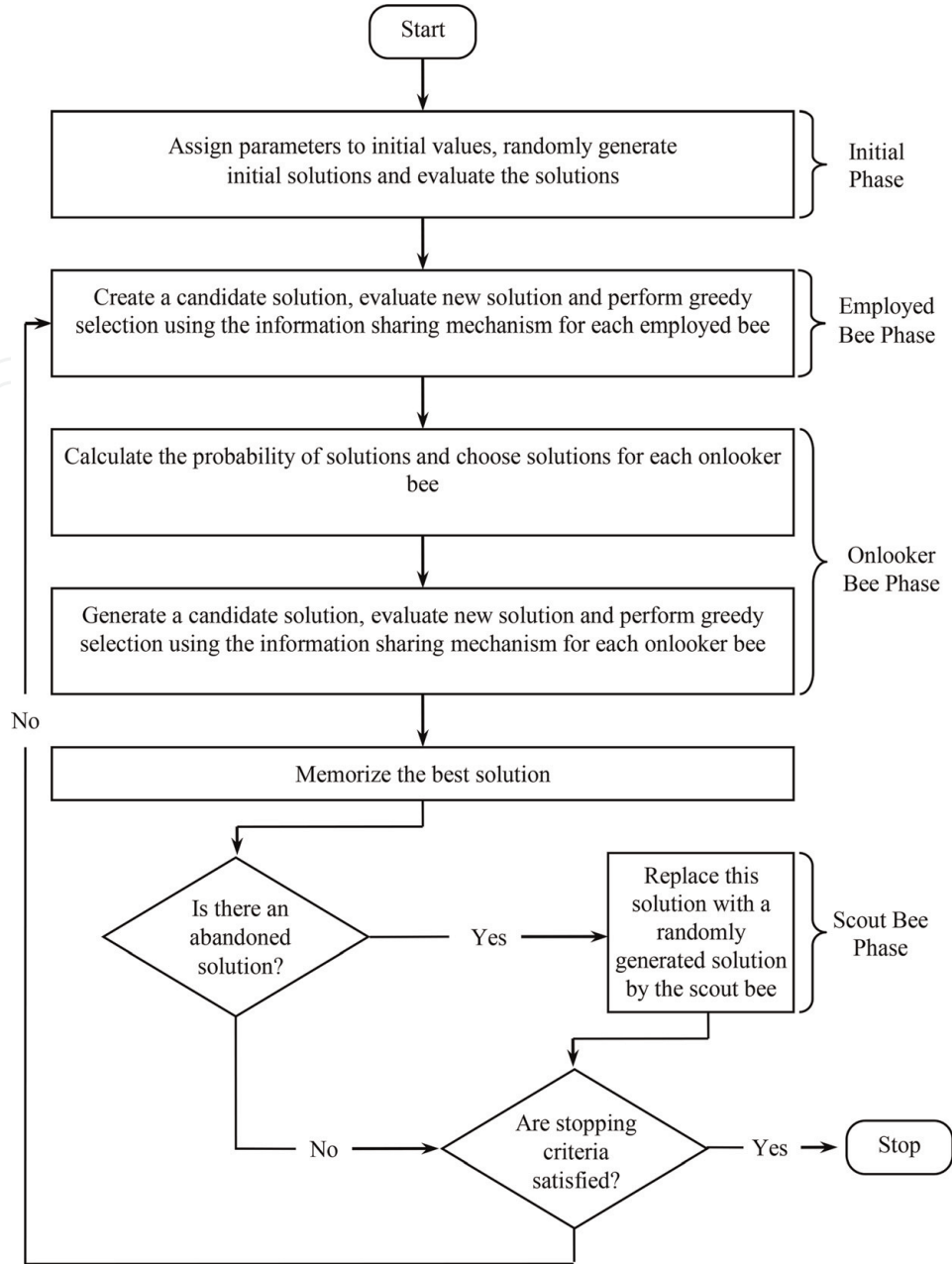*GP and ABCP solutions are represented by tree structure.*

Figure 3.
The flow chart of ABCP.

the nodes of the current solution node x_i and the neighbor solution node x_k. This sharing mechanism is shown in **Figure 4**. **Figure 4a** and **b** are: node x_i representing the current solution and neighbor node x_k taken from the tree respectively, **Figure 4c** neighboring information and the generated candidate solution are given in **Figure 4d**. After the candidate solution is generated, a greedy selection process is applied between the node x_i expressing the current solution and the candidate solution v_i. Candidate solution is evaluated and greedy selection is used for each employed bee.

In onlooker bee phase, employed bees come into hive and share their nectar with the onlooker bees after they complete the research process. The source selection is based on the selection probability of the solution that is based on the nectar qualities, p_i is calculated Eq. (2):

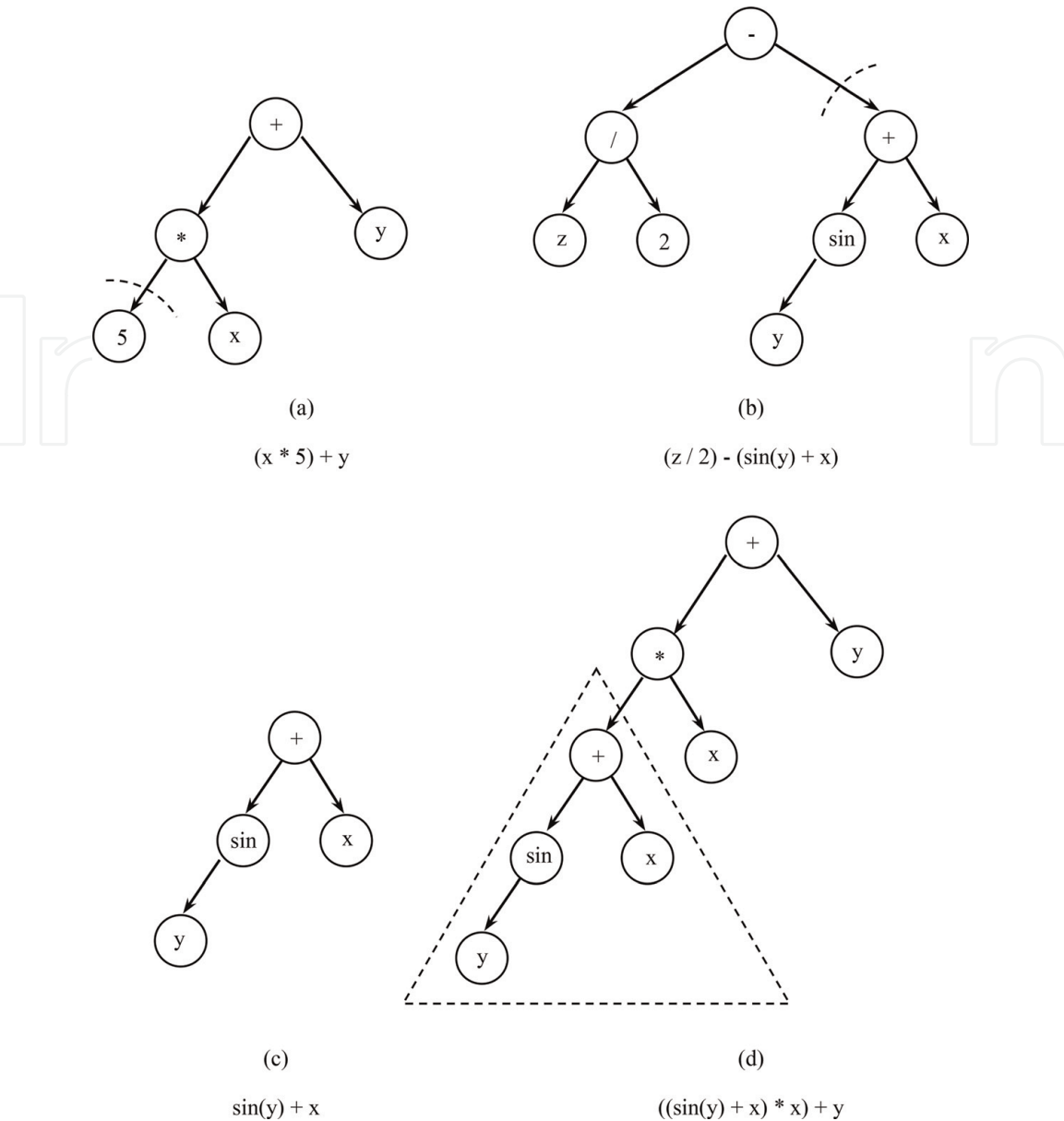$$p_i = \frac{0.9 * fit_i}{fit_{best}} + 0.1 \qquad (2)$$

Figure 4.
Example of information sharing mechanism in ABCP.

where fit_i quality of the solution i, fit_{best} quality of the best solution current solutions [35]. When the solutions are selected, the onlooker bees begin to look for new sources by acting like employed bees. The quality of the newly found solution is checked. If a new solution is more qualified, the solution is taken into memory and the current source is deleted from the memory.

After the employed bees and onlooker bees complete the search in each cycle, the penalty points of the respective sources are incremented by one if they cannot find more qualify sources. When a better source is found, the penalty point of that source is reset. If the penalty point exceeds the 'limit' parameter, the employed bee of that source becomes a scout bee and randomly determines new source instead of an abandoned source.

4. Experimental design

This section demonstrate feature selected classification ability of GP and ABCP, set of experiments conducted.

4.1 Datasets

In this chapter, the experiments are conducted on four real world datasets. All datasets are taken from UCI [37]. The first of data set is Wisconsin diagnostic breast cancer (WDBC). The dataset classifies a tumor as either benign or malignant is the diagnosis of breast cancer. It consists of 30 input parameters that determine whether the tumor of 569 patients is benign or malignant. When the data set is examined, it is observed that ~60% of the benign and remainder of the tumors is malignant. The malignant tumor in the data set is defined as 1 and benign tumor is 0. The entry set contains 10 parameters for the suspected community. These input parameters are given as radius, texture, circumference, area, fluency, density, concavity, concavity points, symmetry and fractal. Dataset has an average, standard error, and worst error value for each record. Thus, there are totally 30 input parameters.

It has been used in much recent work on cancer classification of machine learning algorithms [38–40]. Bagui et al. tried to classify two large breast cancer data sets with many machine learning methods such as linear, quadratic, k-NN [39]. In the paper, 9 variable WBC (Wisconsin breast cancer) and 30 variable WDBC (Wisconsin diagnostics breast cancer) data sets were reduced to 6 and 7 variables, respectively. WDBC is classified J48 decision trees, multi-layer perception (MLP), naive Bayes (NB), sequential minimal optimization (SMO), distance based K nearest neighbor (IBK, instance based for K-nearest neighbor) in [40]. Kathija et al. used support vector machines (SVM) and Naive Bayes to classify WDBC in the paper [40].

The second dataset is the dermatology data set, contains 34 features, 33 of which are linear values and one of which is nominal. The differential diagnosis of erythematosquamous disease is a real problem in dermatology. Diagnosis usually requires a biopsy, but unfortunately, these diseases share many histopathological features. Patients were initially evaluated clinically in the data set. Then, skin samples were taken for evaluation of 22 histopathological features. The values of the histopathological features were determined by analysis of the samples under a microscope. There are multiple researches to diagnose dermatological diseases [41–46]. Rambhajani et al. used the Bayesian technique as a feature selection in the paper [42]. When several measures such as accuracy, sensitivity, and specificity are evaluated high successful results obtained in the model classification of 15 features for the dermatology data set with 34 features. Pappa et al. proposed a multi object GA called C4.5 that performed on six different data sets including the dermatology dataset for feature selection [46].

The other dataset is Wine which is the results of chemical analyzes of wines from three different varieties of the same region of Italy. The analysis is based on the amounts of 13 features present in each of the three wine varieties. Zhong et al. proposed a modified approach to the nonsmooth Newton method and compared with support vector algorithm called standard v-KSVCR method in wine dataset [47]. A proposed block based affine matrix for spectral clustering methods was compared with 10 different datasets including wine dataset standard classification methods in [48].

The last dataset Horse colic which reveals the presence or absence of colic disease depending on various pathological values of horse colic. Nock et al. used the symmetric nearest neighbor (SRN), which calculates the scores of the closest neighbor's relations in [49].

This chapter aims to be able to diagnose that the tumor is benign or malignant in WDBC, to identify six different dermatologic diseases in Dermatology, to recognize

three varieties of wines in Wine and to presence of colic disease was investigated in Horse Colic.

4.2 Training sets and test sets

In this chapter, each dataset is split into a training set and test set to investigate feature selected classification performance of the evolved models. The number of features, training instances and test instances of the four datasets are shown in **Table 1**. All datasets are almost split with 70% of instances randomly selected from the datasets for training and the other 30% instances forms test set. In each run, the training and test instances are reconstructed by selecting from random instances of datasets.

4.3 Settings

Similar parameter values and functions are used for comparison with GP and ABCP. Since the real input features of the data sets were used, the results obtained from the solutions are theoretically in the range of $[-\infty, \infty]$. Result values to be able to define discrete class values (such as class 0, class 1), it is necessary to be first drawn to a range defined earlier and be contained the total number of classes. The fitness function is defined in Eq. (3).

$$(N_c - 1) * \left(\frac{1}{(1 + \exp(-g_0))} \right) \tag{3}$$

where N_c is the number of output classes, g_o is the result of the current solution. For example, for a problem of class 4, the output of Eq. (3) is in the range $[0–3]$. The real features found are rounded to the nearest integer value and the solution class features are predicted as '0', '1', '2', '3' in this case.

In this chapter, the fitness function is the weighted sum of the ratios of the total class numbers in the data set of correctly predicted class numbers. For example, in the binary classification, the fitness function is obtained by summing up ratio of correct predicted 0 to total number of 0 in the data set with ratio of correct predicted 1 to total number of 1 in the data set.

For binary classification problems, this function is defined as SFF (sensitivity fitness function) given in Eq. (4) [50].

$$SFF = w \frac{n_c(i, 0)}{n_a(i, 0)} + (1 - w) \frac{n_c(i, 1)}{n_a(i, 1)} \tag{4}$$

Dataset	Features	Total instances	Training instances	Test instances	Output classes
WDBC	30	569	427	142	2
Dermatology	34	366	274	92	6
Wine	13	178	133	45	3
Horse colic	26	364	273	91	3

Table 1.
Characteristics of the datasets considered in the experiments.

where $n_c(i,k)$ is the number of correctly predicted states when compared to the k class in data set from the class k for the ith solution, $n_a(i,k)$ the number of all records in class k in the data set is the number of inputs defined in the range [0, 1] refers to a real number. The generalized version of Eq. (4) is given in Eq. (5) for multiple class problems investigated.

$$SFF_n = \sum_{j=0}^{n-1} w \frac{n_c(i,j)}{n_a(i,j)} \tag{5}$$

In general, the weight value (w) is used equally. In this case, the proportion of the ratio distribution for each class is adjusted equally. In some cases, a penalty parameter can be added to avoid misclassification in unbalanced data sets. The parameter is added to the fitness function that defined in Eq. (5) as expressed in Eq. (6). It evaluates the models obtained from the solutions. Where p is the penalty factor and N is the total number of nodes in the solution.

$$SFF_n = \sum_{j=0}^{n-1} w \frac{n_c(i,j)}{n_a(i,j)} - pN \tag{6}$$

The data sets are evaluated according to the SFF function defined in Eq. (6). The complexity of the obtained solution is calculated as in Eq. (7) in proportion to the depth of the tree and the number of nodes.

$$C = \sum_{k=1}^{d} n * k \tag{7}$$

where C is tree complexity, d is the depth of the solution tree, and n is the number of nodes at depth.

The control parameters used by the automatic programming methods are given in **Table 2**. The population size and the iteration size are set by the number of features and the number of classes of the data set. Dermatology has more features and classes than other datasets, therefore population size and iteration number are

Control parameters	WDBC		Dermatology		Wine		Horse colic	
	GP	ABCP	GP	ABCP	GP	ABCP	GP	ABCP
Population/colony size	200	200	300	300	300	300	300	300
Iteration size	150	150	250	250	150	150	250	250
Maximum tree depth	12	12	12	12	12	12	12	12
Tournament size	6	—	6	—	6	—	6	—
Mutation ratio	0.1	—	0.1	—	0.1	—	0.1	—
Crossover ratio	0.8	—	0.8	—	0.8	—	0.8	—
Direct reproduction ratio	0.1	—	0.1	—	0.1	—	0.1	—
w	1/2		1/6		1/3		1/3	
p	0.001		0.001		0.001		0.001	
Functions	+, −, $*$, tan, sin, cos, square, maxx, minx, exp., ifbte, iflte							

Table 2.
Control parameters of GP and ABCP in the experiments.

chosen as the highest for this dataset. As seen from **Table 2**, the weight value is defined in proportion to the number of classes in the output of each data set. Each class is equal importance. The penalty point given in Eq. (6) was set equal to 0.001 for all data sets. The *maxx* function specifies the maximum value of vector, the *minx* function specifies the minimum value of vector. The *ifbte* checks the value of left operand, if it is greater than or equal to the value of right operand, then condition becomes true. The *iflte* checks the value of left operand, if it is less than or equal to the value of right operand, then condition becomes true. How the functions operate condition expressions are defined in Eqs. (8) and (9).

$$X = ifbte(A, B, C, D)$$
$$if (A \geq B) \; then \;\; X = C \; else \; X = d$$

(8)

$$X = iflte(A, B, C, D)$$
$$if (A < B) \; then \;\; X = C \; else \; X = d$$

(9)

4.4 Simulation results

For each data set, GP and ABCP are run 30 times according to configuration in **Table 2**. The classification success of GP and ABCP methods are given in **Table 3** in terms of mean, best and worst values for each dataset. SFF and success percentage (SP) results are given in **Table 3** for both training and test cases. As the SFF increased, the success rate of classification increased. The highest mean classification in training (93.43%) was obtained ABCP in Wine. Both methods showed lower SFF and classification success compared to other data sets in Horse colic. The best

Dataset	Metrics	GP				ABCP			
		Train		Test		Train		Test	
		SFF	SP	SFF	SP	SFF	SP	SFF	SP
WDBC	Mean	0.91	92.33	0.9	91.01	0.92	93.27	0.9	91.48
	Standard deviation	0.02	2.56	0.03	3.8	0.02	2.01	0.03	3.07
	Best	0.94	95.32	0.94	95.77	0.95	96.25	0.96	97.89
	Worst	0.86	86.42	0.81	77.46	0.87	87.82	0.84	84.51
Dermatology	Mean	0.81	81.96	0.77	78.66	0.89	92.27	0.85	89.17
	Standard deviation	0.1	15	0.11	13.96	0.02	1.93	0.05	4.4
	Best	0.92	95.26	0.94	96.74	0.93	97.08	0.97	98.91
	Worst	0.6	48.54	0.48	46.74	0.84	89.42	0.77	80.43
Wine	Mean	0.88	88.7	0.85	84.9	0.92	93.43	0.88	88.22
	Standard deviation	0.06	5.94	0.07	7.59	0.02	2.59	0.05	6.83
	Best	0.95	98.5	0.98	100	0.97	98.5	0.98	100
	Worst	0.76	76.69	0.71	73.33	0.88	88.72	0.78	73.33
Horse colic	Mean	0.62	58.81	0.49	50.4	0.67	62.52	0.54	54.76
	Standard deviation	0.06	5.42	0.09	8.35	0.03	3.53	0.07	4.92
	Best	0.71	67.4	0.65	71.43	0.73	69.96	0.65	61.54
	Worst	0.51	47.99	0.3	38.46	0.62	56.78	0.36	45.05

Table 3.
Classification results for each data set.

Dataset	Method	Model of best of run individual	Number of features
WDBC	ABCP	$\cos(x_8 * x_{22}) + \min(x_5, \cos(x_5 * x_{21}))$	4
	GP	$\cos(\textit{iflte}(\tan(0.7604 * x_{27} * x_{28} * x_{24}), \tan(x_{14}), \tan(0.7390), \cos(x_{17})))$	5
Dermatology	ABCP	$(\textit{ifbte}(\tan(x_{15}, x_6, (((\tan(\textit{ifbte}(x_{22}, x_{22}, (\textit{ifbte}(x_{33}, x_{31}, x_{31}, \textit{ifbte}(x_{23}, x_{31}, \tan(-2.161), x_{15})) - x_{31}, x_{15}))) - x_{22}) + (\tan(x_5 * x_{34}))), (2x_7 - x_{20})) - x_{22})$	11
	GP	$\textit{ifbte}(x_{13}, x_{20}, \textit{ifbte}(x_{13}, x_{27}, \cos(\exp(\textit{ifbte}(x_{23}, x_5, \cos(x_{15}), x_{31}), x_{23}), (x_{13} - x_{23})))) + (x_{31} - (x_{22} * \exp(x_{22})))$	8
Wine	ABCP	$\textit{ifbte}(-2.452, (\exp(x_{12}) - x_{10}), 7.5519, ((-2.452 * x_{10}) * \cos(x_7)) * \min(\cos(\cos(x_{10} - x_7)) - x_{13})))$	4
	GP	$\min(\exp(\sin(x_1), \min(\exp(\cos(x_7)), \cos(\textit{iflte}(x_9, x_7, \textit{iflte}(\cos(\exp(\sin(x_1))), \textit{iflte}(\cos(x_7), x_8, x_7, x_8))x_7,)x_{11})), \tan(\textit{iflte}(x_{11}, \sin(x_1), x_8, x_7)))$	4
Horse colic	ABCP	$(\textit{ifbte}(\textit{square}(\cos(x_{10} + x_{24})), \sin(x_{22}), \min x(x_8, x_{12}) - 4.1591, \sin(\textit{ifbte}(x_{19}, \exp(\exp(\textit{iflte}(x_{23}, x_{22}, x_{22}, \sin(x_{22}))), \tan(x_{23}) - x_{22}, x_{22}))))$	7
	GP	$\min(x_6, x_{21}) - \textit{square}(\min(\textit{iflte}(x_{22}, x_{10}, \textit{iflte}(\textit{square}(x_1), x_{10}, \tan(-6.653), x_1), x_{26}), \min(x_{14}, x_{21}) - \textit{square}(\min(\textit{iflte}(\textit{square}(x_{14}), \textit{square}(x_8), \tan(x_{23}), -6.655), (x_{23} - x_1)))))$	8

Table 4.
Models of best run ABCP and GP.

models of GP and ABCP have 100% test classification success in Wine. For the case study investigated, compact classification models are obtained with comparable accuracy to GP.

Problem	GP			ABCP		
	Total number of nodes	Depth of the best solution tree	Best solution tree complexity	Total number of nodes	Depth of the best solution tree	Best solution tree complexity
WDBC	16	7	67	**11**	5	**36**
Dermatology	**25**	**8**	**107**	37	12	249
Wine	32	9	177	**21**	7	**81**
Horse colic	34	9	197	**33**	9	**163**

Table 5.
Best solution tree information for each data set.

Program	Metrics	Mean	Standard deviation	Most common features	Features in both GP and ABCP	Number most common features	Number features both GP and ABCP
WDBC	ABCP	4.13	1.33	$x_{28}(15), x_7(12),$ $x_8(11), x_5(7)$	$x_{28}(15), x_7(12),$ $x_8(11)$	4	3
	GP	3.13	1.36	$x_8(12), x_7(12),$ $x_{27}(11), x_{28}(8)$	$x_8(12), x_7(12),$ $x_{28}(8)$	4	3
Dermatology	ABCP	7.20	1.90	$x_{31}(30), x_{15}(29),$ $x_{22}(25), x_{14}(23),$ $x_{33}(15), x_7(12),$ $x_{27}(10), x_6(9)$	$x_{31}(30), x_{15}(29),$ $x_{22}(25), x_{14}(23),$ $x_{33}(15), x_7(12),$ $x_{27}(10)$	8	7
	GP	6.23	1.74	$x_{31}(23), x_{22}(15),$ $x_{14}(15), x_7(13),$ $x_5(10), x_{20}(10),$ $x_{27}(9), x_{15}(9),$ $x_{30}(8), x_{33}(8)$	$x_{31}(23), x_{22}(15),$ $x_{14}(15), x_7(13),$ $x_{27}(9), x_{15}(9),$ $x_{33}(8)$	10	7
Wine	ABCP	4.07	1.18	$x_7(30), x_{11}(26),$ $x_{10}(19), x_{12}(17)$	$x_7(30), x_{11}(26),$ $x_{10}(19), x_{12}(17)$	4	4
	GP	3.17	1.58	$x_7(29), x_{10}(14),$ $x_{12}(12), x_{11}(12)$	$x_7(29), x_{10}(14),$ $x_{12}(12), x_{11}(12)$	4	4
Horse colic	ABCP	6.93	1.41	$x_{23}(27), x_{19}(25),$ $x_{22}(23), x_1(21),$ $x_{21}(13), x_{26}(13),$ $x_8(13), x_{15}(7),$ $x_{10}(7)$	$x_{23}(27), x_{19}(25),$ $x_{22}(23), x_1(21),$ $x_{21}(13), x_{26}(13),$ $x_8(13), x_{10}(7)$	9	8
	GP	5.97	2.36	$x_{23}(15), x_1(15),$ $x_{21}(14), x_{26}(14),$ $x_{19}(14), x_8(13),$ $x_{22}(12), x_7(11),$ $x_{14}(9), x_{10}(9),$ $x_2(7)$	$x_{23}(15), x_1(15),$ $x_{21}(14), x_{26}(14),$ $x_{19}(14), x_8(13),$ $x_{22}(12), x_{10}(9)$	11	8

Table 6.
Number of features selected by the methods.

4.5 Analysis of evolved models

The evolved models of best classifier solutions in ABCP are shown in **Table 4**. It can be observed that both methods extracted successful models with few features. The methods extracted models regardless of the total number of features of the data sets. In general, ABCP has achieved higher success rate of classification than GP using less features.

Table 5 shows general information about the best solution tree. Less complex models are shown in the table with bold typing. When the trees of the best models are analyzed structurally, ABCP, except for the dermatology, shows the best models with less complexity. The detailed information about the inputs of mathematical models of the best solutions in each run are presented in **Table 6**. Features are ordered most common in equations on the table. Equations which are most common, three features (x_7, x_8, x_{28}) are same in WDBC; seven features (x_7, x_{14}, x_{15}, x_{22}, x_{27}, x_{31}, x_{33}) are same in dermatology; four features (x_7, x_{10}, x_{11}, x_{12}) are same in wine; eight features (x_1, x_8, x_{10}, x_{19}, x_{21}, x_{22}, x_{23}, x_{26}) are same in horse colic in both methods. In the best models of the 30 runs, the frequently available features in both of the methods were evaluated as inputs for success of classification. For example, in total 30 runs for WDBC x_{28} 15 times; for dermatology x_{31} were seen.

5. Conclusion

In this chapter, selecting features in classification problems are investigated using GP and ABCP and the literature study related to this field is included. In the performance analysis of the methods, four classification problems are used. As results of 30 runs, the features of the best models were examined. Both methods were found to extract successful models with the same features. According to the experimental results, ABCP is able to extract successful models in training set and it has comparable accuracy to GP. This chapter shows that ABCP can be used in high level automatic programming for machine learning. Several interesting automatic programming methods such as Multi-Gen GP and Multi-Hive ABCP can be further researched in the near future.

Author details

Sibel Arslan and Celal Ozturk*
Computer Engineering Department, Engineering Faculty, Erciyes University, Kayseri, Turkey

*Address all correspondence to: celal@erciyes.edu.tr

IntechOpen

References

[1] Nag K, Pal NR. A multiobjective genetic programming based ensemble for simultaneous feature selection and classification. IEEE Transactions on Cybernetics. 2016;**46**:499-510. DOI: 10.1109/TCYB.2015.2404806

[2] Yu J, Yu J, Almal AA, Dhanasekaran SM, Ghosh D, Worzel WP, et al. Feature selection and molecular classification of cancer using genetic programming. Neoplasia. 2007;**9**(4):292-303. DOI: 10.1593/neo.07121

[3] Zhang Y, Rockett PI. Domain-independent feature extraction for multi-classification using multi-objective genetic programming. Pattern Analysis and Applications. 2010;**13**(3): 273-288. DOI: 10.1007/s10044-009-0154-1

[4] Muni DP, Pal NR, Das J. Genetic programming for simultaneous feature selection and classifier design. IEEE Transactions on Systems, Man, and Cybernetics. 2006;**36**(1):106-117. DOI: 10.1109/TSMCB.2005.854499

[5] Cai R, Hao Z, Yang X, Wen W. An efficient gene selection algorithm based on mutual information. Neurocomputing. 2009;**72**:91-999. DOI: 10.1016/j.neucom.2008.04.005

[6] Saeys Y, Inza I, Larranaga P. Review of feature selection techniques in bioinformatics. Bioinformatics. 2007; **23**(19):2507-2517. DOI: 10.1093/bioinformatics/btm344

[7] Xue B, Zhang M, Browne WN, Yao X. A survey on evolutionary computation approaches to feature selection. IEEE Transactions on Evolutionary Computation. 2016;**20**(4): 606-626. DOI: 10.1109/TEVC.2015.2504420

[8] Guyon I, Elisseeff A. An introduction to variable and feature selection. Journal of Machine Learning Research. 2003;**3**: 1157-1182

[9] Gulgezen G. Kararlı ve başarımı yüksek öznitelik seçimi. Istanbul Technical University; 2009

[10] Zhang Y, Wanga S, Phillips P, Ji G. Binary PSO with mutation operator for feature selection using decision tree applied to spam detection. Knowledge-Based Systems. 2014;**64**:22-31. DOI: 10.1016/j.knosys.2014.03.015

[11] Sikora R, Piramuthu S. Framework for efficient feature selection in genetic algorithm based data mining. European Journal of Operational Research. 2007; **180**:723-737. DOI: 10.1016/j.ejor.2006.02.040

[12] Dy JG, Brodley CE. Feature selection for unsupervised learning. Journal of Machine Learning Research. 2004;**5**: 845-889

[13] Schiezaro M, Pedrini H. Data feature selection based on artificial bee colony algorithm. EURASIP Journal on Image and Video Processing. 2013;**47**:1-8

[14] Landry JA, Costa LD, Bernier T. Discriminant feature selection by genetic programming: Towards a domain independent multiclass object detection system. Systemics Cybernetics and Informatics. 2006;**3**(1):7681

[15] Fidelis MV, Lopes HS, Freitas AA. Discovering comprehensible classification rules with a genetic algorithm. In: IEEE, Proceedings of the Congress; Vol. 1. 2000. pp. 805-810. DOI: 10.1109/CEC.2000.870381

[16] Athitsos V, Sclaroff S. Boosting nearest neighbor classifiers for multiclass recognition. In: Computer Science Tech Report; 2004. DOI: 10.1109/CVPR.2005.424

[17] Venkatesan R, Er MJ. A novel progressive learning technique for multiclass classification. Neurocomputing. 2016;**207**:310-321. DOI: 10.1016/j.neucom.2016.05.006

[18] Awuley A, Ross BJ. Feature selection and classification using age layered population structure genetic programming. In: CEC 2016; 2016. DOI: 10.1109/CEC.2016.7744088

[19] Lin JY, Ke HR, Chien BC, Yang WP. Classifier design with feature selection and feature extraction using layered genetic programming. Expert Systems with Applications. 2008;**34**(2): 1384-1393. DOI: 10.1016/j.eswa.2007. 01.006

[20] Ahmed S, Zhang M, Peng L. Feature selection and classification of high dimensional mass spectrometry data, a genetic programming approach. In: Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics. 11th European Conference EvoBIO 2013. Vienna, Austria; 2013. pp. 43-55. DOI: 10.1007/ 978-3-642-37189-9_5

[21] Liu KH, Tong M, Xie ST, Yee VT. Genetic programming based ensemble system for microarray data classification. Computational and Mathematical Methods in Medicine. Hindawi Publishing Corporation. 2015; **2**:1-11. DOI: 10.1155/2015/193406

[22] Karaboga D, Ozturk C. A novel clustering approach: Artificial bee colony (ABC) algorithm. Applied Soft Computing. 2011;**11**:652-657. DOI: 10.1016/j.asoc.2009.12.025

[23] Karaboga D, Ozturk C. Neural networks training by artificial bee colony algorithm on pattern classification. Neural Network World: International Journal on Neural and Mass Parallel Computing and Information Systems. 2009;**19**(3): 279-292

[24] Joyanth J, Kumar A, Koliwad S, Krishnashastry S. Artificial bee colony algorithm for classification of remote sensed data. In: Industrial Instrumentation and Control (ICIC), International Conference. 2015. DOI: 10.1109/IIC.2015.7150989

[25] Chung YY, Yeh W, Wahid N, Mujahid A, Zaidi A. Artificial bee colony based data mining algorithms for classification tasks. Modern Applied Science. 2011;**5**(4):217-231. DOI: 10.5539/mas.v5n4p217

[26] Koza J. Genetic Programming: On the Programming of Computers by Means of Natural Selection. Cambridge, MA, USA: MIT Press; 1992

[27] Koza J. Genetic Programming II: Automatic Discovery of Reusable Programs. Cambridge, MA: MIT Press; 1994

[28] Koza J, Bennett F, Andre D, Keane M. Genetic Programming III: Darwinian Invention and Problem Solving. IEEE Transactions on Evolutionary Computation. San Francisco, CA; **3**(3):251-253

[29] Zhang L, Nandi AK. Fault classification using genetic programming. Mechanical Systems and Signal Processing. 2007;**21**(3): 1273-1284. DOI: 10.1016/j.ymssp.2006. 04.004

[30] Settea S, Boullartb L. Genetic programming: Principles and applications. Engineering Applications of Artificial Intelligence. 2001;**14**:727-736. DOI: 10.1016/S0952-1976(02)00013-1

[31] Poli R, Langdon W, McPhee N. A Field Guide to Genetic Programming. England, UK; 2008:19-27. http://lulu. com, Creative Commons Attribution, Noncommercial-No Derivative Works 2.0

[32] Gan Z, Chow TWS, Chau WN. Clone selection programming and its

application to symbolic regression. Expert Systems with Applications. 2009;**36**:3996-4005. DOI: 10.1016/j. eswa.2008.02.030

[33] Karaboga D. Yapay Zeka Optimizasyon Algoritmaları. Nobel Yayınları; 2011

[34] Karaboga D. An Idea Based On Honey Bee Swarm for Numerical Optimization. Technical Report TR06. Erciyes University, Engineering Faculty, Computer Engineering Department; 2005

[35] Karaboga D, Ozturk C, Karaboga N, Gorkemli B. Artificial bee colony programming for symbolic regression. Information Sciences. 2012;**209**:115. DOI: 10.1016/j.ins.2012.05.002

[36] Gorkemli B. Yapay Arı Koloni Programlama (ABCP) yöntemlerinin geliştirilmesi ve sembolik regresyon problemlerine uygulanması, PhD Thesis, Erciyes University, Engineering Faculty, Computer Engineering Department; 2015

[37] UC Irvine Machine Learning Repository. [Online]. Available from: http://archive.ics.uci.edu/ml/index.php

[38] Bagui S, Bagui S, Hemasinha R. The statistical classification of breast cancer data. International Journal of Statistics and Applications. 2016;**6**(1):15-22. DOI: 10.5923/j.statistics.20160601.03

[39] Salama GI, Abdelhalim MB, Zeid MA. Breast cancer diagnosis on three different datasets using multiclassifiers. International Journal of Computer and Information Technology. 2012;**01**:2277-0764

[40] Kathija A, Nisha S. Breast cancer data classification using SVM and naive Bayes techniques. International Journal of Innovative Research in Computer and Communication Engineering. 2016;**4**:12

[41] Guvenir HA, Demiröz G, Ilter N. Learning differential diagnosis of erythematosquamous diseases using voting feature intervals. Artificial Intelligence in Medicine. 1998;**13**: 147-165

[42] Rambhajani M, Deepanker W, Pathak N. Classification of dermatology diseases through Bayes net and best first search. International Journal of Advanced Research in Computer and Communication Engineering. 2015;**4**(5): 116-119. DOI: 10.17148/IJARCCE.2015. 4526

[43] Manjusha K, Sankaranarayanan K, Seena P. Data mining in dermatological diagnosis: A method for severity prediction. International Journal of Computers and Applications. 2015; **117**(11):0975-8887

[44] Barati E, Saraee M, Mohammadi A, Adibi N, Ahamadzadeh MR. A survey on utilization of data mining approaches for dermatological (skin) diseases prediction. Cyber Journals: Multidisciplinary Journals in Science and Technology. Journal of Selected Areas in Health Informatics (JSHI). March Edition, 2011:1-11

[45] Parikh KS, Shah TP, Kota R, Vora R. Diagnosing common skin diseases using soft computing techniques. International Journal of Bio-Science and Bio-Technology. 2015;7(6):275-286. DOI: 10.1109/ICASTECH.2009.5409725

[46] Pappa GL, Freitas AA, Kaestner CAA. Attribute selection with a multi objective genetic algorithm. In: SBIA; 2002

[47] Zhong P, Fukushima M. A regularized non-smooth newton method for multiclass support vector machines. Optimization Methods and Software. 2007;**22**:225-236. DOI: 10.1080/ 10556780600834745

[48] Fischer I, Poland J. Amplifying the block matrix structure for spectral

clustering. Technical Report No. IDSIA0305; 2005

[49] Nock R, Sebban M, Bernard D. A simple locally adaptive nearest neighbor rule with application to pollution forecasting. International Journal of Pattern Recognition and Artificial Intelligence. 2003;**17**(8):1369-1382. DOI: 10.1142/S0218001403002952

[50] Morrison GA, Searson DP, Willis MJ. Using genetic programming to evolve a team of data classifiers. International Journal of Computer, Electrical, Automation, Control and Information Engineering. 2010;**4**(72): 261-264