Journal of Theoretical and Applied Information Technology
31st January 2019. Vol.97. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

INTELLIGENT CLASSIFICATION ALGORITHMS IN ENHANCING THE PERFORMANCE OF SUPPORT VECTOR MACHINE

[1]HIBA BASIM ALWAN, [2]KU RUHANA KU-MAHAMUD
[1,2]School of Computing, Universiti Utara Malaysia, 06010 Sintok, Kedah, Malaysia
[1]hiba81basim@yahoo.com, [2]ruhana@uum.edu.my

ABSTRACT

Performing feature subset and tuning support vector machine (SVM) parameter processes in parallel with the aim to increase the classification accuracy is the current research direction in SVM. Common methods associated in tuning SVM parameters will discretize the continuous value of these parameters which will result in low classification performance. This paper presents two intelligent algorithms that hybridized between ant colony optimization (ACO) and SVM for tuning SVM parameters and selecting feature subset without having to discretize the continuous values. This can be achieved by simultaneously executing the selection of feature subset and tuning SVM parameters simultaneously. The algorithms are called ACO_{MV}-SVM and $IACO_{MV}$-SVM. The difference between the algorithms is the size of the solution archive. The size of the archive in ACO_{MV} is fixed while in $IACO_{MV}$, the size of solution archive increases as the optimization procedure progress. Eight benchmark datasets from UCI were used in the experiments to validate the performance of the proposed algorithms. Experimental results obtained from the proposed algorithms are better when compared with other approaches in terms of classification accuracy. The average classification accuracies for the proposed ACO_{MV}–SVM and $IACO_{MV}$-SVM algorithms are 97.28 and 97.91 respectively. The work in this paper also contributes to a new direction for ACO that can deal with mixed variable ACO.

Keywords: *Support Vector Machine, Ant Colony Optimization, Parameter Optimization, Feature Subset Selection, Evolutionary Approach*

1. INTRODUCTION

With the continuing growth of intelligent approaches, model of intelligent computing methods in optimization application such as particle swarm algorithm, ant colony algorithm, differential evolution algorithm and genetic algorithm have increased significantly. These intelligent methods are often based on population probability search and will not fall into local extremism. Thus, these intelligent methods can solve the limitations of traditional calculation method and improve the accuracy and efficiency of optimization problem [1].

Artificial intelligence has been utilized in expert system, pattern recognition, machine learning, classification and clustering. Machine Learning (ML) is the improvement of approaches that permit computers to learn built on experimental data. The aim of ML is to construct computer systems that familiarize and learn from their knowledge. Machine learning is able to be either unsupervised or supervised. Classification is one example of supervised learning which expressed as the job of learning from inputs represented through a set of attributes and a class label. The output of learning is a model of classification that is able to forecast the class label of unlabelled inputs [2]. Over the past decades, ML algorithms have present significant ability in solving problems from various pools [3]. Different approaches like decision tree, artificial neural networks, support vector machine (SVM), nature-inspired methods like genetic programming and instance based learning methods have been suggested in literature for classification [2]. Through different approaches, SVM has obtained big reputation because of its theoretical foundation and approximately higher behaviour as compared to reset learning methods based on different practical classification problems [3]. Selecting the best kernel function is important for SVM and the often favoured kernel function is the Radial Basis Function (RBF). Radial Basis Function provides optimal result if the variables are correctly selected [2].

Pattern classification is an important component in the decision-making process for any intelligent system. Prediction of unknown samples using a learned classifier is known as pattern classification. Some of the popular classifiers include SVM, neural networks, decision trees, nearest neighbour and linear discriminant analysis [4]. An increasing interest is detected in the machine learning research society regarding the structure of algorithms and purposes that consider both the confidentiality of the examples to be manipulated and the reliability of the events contained in the operation [5].

In SVM, a binary classification problem is constructed like a convex Quadratic Problem Program (QPP) that has a single universe solution. The solution of QPP needs a great calculation fee $O(m^3)$, where training examples is m. The decision limit is expressed through a hyper-plane in the attributes domain [3]. Support vector machine which is a supervised ML approach, classifies the pattern through training and testing of data, plays a vital role in pattern classification. Support vector machine aims to generate a prototype which forecast the goal value of input features in the testing subset. Target values or recognized labels pointer if the approach is operate satisfactorily or not, which examples to a target value, confirming the precision of the approach, or be utilized to assist the approach to learn and perform in a desired way [6]. The main concept of SVM is to obtain the optimal isolating hyper-plane through the + and - instances. The optimum hyper-plane can be found through increasing the margin through two adjusts hyper-planes, which involve the decreasing of a QPP. Through presenting kernel fake into twin QPP, SVM may also resolve non-linear classification problems correctly [7]. Subsequently SVM tracks the structural risk minimization concept which will minimize error through the training stage and improves its generalization ability. Because of its great behaviour, it has since been highly globalized and implemented to various problems [8]. The behaviour of SVM is generally counted on its variables and feature subset which is consider as two necessary factors for enhancing SVM behaviour and are classically resolved in isolation. Utilizing either variable parameter optimization or chosen feature is bad in the aim to obtain optimal behaviour. These two problems have influenced on each other. Thus to obtain good classification behaviour, choosing the best attributes subset and SVM model selection should be performed simultaneously, as both fit to the combinatorial optimization problem. Furthermore, both problems could be managed with swarm intelligence and evolutionary approaches [9].

Solving real optimization problems is complex for big-scale problems in terms of computational time. The modelling of these problems is also tedious. Metaheuristics are efficient approaches to find satisfactory solution in good time and can produce generic approach framework that may be utilized in different problems with little changes [10].

Ant colony optimization (ACO) is a metaheuristics Swarm Intelligent (SI) algorithm based on the foraging activities of ants which has been applied in solving discrete optimization problems [11]. Artificial ants build solutions uncountable and conduct with each other through a stigmergy technique. This procedure is performed iteratively till a halt condition is encountered. The main important attribute of ACO is the positive feedback that takes advantages from the pheromone deposited through ants that may lead to the solution building procedure [10]. It is a community build on stochastic universe seek approach, which was firstly suggested by Dorigo in 1991. The approach has dual mechanisms. The first is built on heuristics and is found from the difficult prototype. The second is to utilize data on pheromone on the way to obtain good solutions [12]. Ant colony optimization variants have been used to resolve discrete optimization problem such as routing, scheduling, classification and clustering [13-17], Ant colony optimization is also appropriate for selecting feature subset problems, but not appropriate in achieving the optimal SVM parameters [9, 18, 19]. However, an enhanced ACO algorithm is considered here for simultaneously optimizing SVM variables and attributes subset.

This paper proposes two intelligent algorithms that can overcome the discretization process of continuous values when tuning the SVM parameters. The ACO variants (ACO_{MV} and $IACO_{MV}$) have been used to identify suitable input feature subset and value for SVM parameters. The processes are executed simultaneously. Feature subsets are selected through a wrapper approach which are then transfer to SVM for classification process. Feature subset selection via the wrapper approach follows the inductive learning approach where feedback is obtained from the classifier and the process of features selection is repeated to obtain better features. The benefit in performing the processes simultaneously is to eliminate the accumulation of error from feature selection phase to tuning SVM parameters phase. Thus better classification accuracy can be obtained. To the knowledge of the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

researcher, there is no study that considers using continuous and mixed-variable ACO based SVM for pattern classification. This study has also covered all the important constraints which make the SVM classifier more accurate compared to previous researches.

The rest of the paper is organized as follows. Related works are discussed in Section 2, while the basic concept of SVM and ACO are presented in Section 3. Section 4 presents the research methodology while the proposed algorithms are presented in Section 5. Experimental results are discussed in Section 6 while Section 7 highlights the conclusion and future work.

2. RELATED WORK

Support vector machine has been utilized to resolve classification problems with acceptable accuracy while simultaneously optimizing both attributes subset selection and SVM variables. The current research direction has moved towards simultaneously optimizing both attributes subset selection and tuning SVM variables using optimization algorithms. This approach will increase the classification accuracy because selecting the suitable feature subset and values for SVM parameters influence each other, and in turn, will influence the classification accuracy [20-33]. Techniques such as Mixed-Variable ACO with continuous ACO (ACO_R) to optimize continuous parameters (ACO_{MV-R}), particle swarm optimization (PSO), ACO, genetic algorithm (GA), Immune Clonal Algorithm (ICA), Cat Swarm Optimization (CSO), Clonal Selection Algorithm (CSA), Real-valued Gravitational Search Algorithm (RGSA), Binary (discrete) GSA (BGSA), and Adaptive Cohort Intelligence (SACI) are used to simultaneously optimize attributes subset selection and value for SVM variables.

A total of fifteen similar works [20-22, 25-36] proposed utilizing hybrid approaches to improve classification precision through utilizing limited, appropriate attributes subsets. All fourteen works optimized attributes subset and SVM variables, which are C and γ RBF kernel variables, simultaneously. Ultimately, SVM is utilized to calculate the quality of the solution for all hybrid approaches. The difference is what the hybrid approaches are built on. The summarization of these works is illustrated in Table 1.

Table 1. Studies on simultaneous optimization of model and feature subset selections

Reference	MS and FS Algorithm	Kernel Function	Dataset
[20]	ACOMV-R-SVM	RBF	UCI
[21]	SACI	RBF	---
[22]	RGSA to optimize MS and BGSA to optimize feature subset selection	RBF	UCI
[25]	GA	RBF	UCI
[26]	ACO	RBF	UCI
[27]	CSA	RBF	UCI
[28]	Mixed discrete and continuous PSO	RBF	Distributed parallel
[29]	GA	RBF	UCI
[30]	CSO	RBF	UCI
[31]	Bees algorithm	RBF	UCI
[32]	Continuous Hide-and-Seek SA to optimize MS and discrete SA to optimize feature subset selection	RBF	UCI
[33]	Discrete PSO	RBF	UCI
[34]	Modified Binary PSO (MBPSO)	RBF	UCI
[35]	ICA	RBF	UCI
[36]	Dual formula SVM involving a penalization function based on 0-norm approximation and modifying Gaussian kernel utilizing gradient descent approximation for kernel optimization and feature deletion	Linear Polynomial Gaussian	---

From previous studies, the UCI datasets were the most used data repository and RBF is the most applied kernel function. ACO_{MV-R}, ACO, GA, PSO ICA, SA, Bees, CSA, CSO, and RGSA were the optimization algorithms that have been utilized to simultaneously optimize feature subset and tuning SVM parameters.

The continuous values of SVM parameters will have to be converted into discrete values before an optimization process is carried out for the case of ACO or other approaches such as GA, PSO, CSO and CSA are hybridized in SVM [23, 26]. The average classification accuracy for the studies is 92.27. This conversion process has resulted in the loss of some information and will affect the classification accuracy [26], because it will restrict the details at which hopeful regions of the seek domain can be investigated. To overcome this problem, new ways to take continuous values of SVM parameters without converting to discrete forms are required.

When ACO is used to select feature subset, the features would be represented as discrete graph nodes, while SVM parameters are naturally continuous [26]. New ways to accept mixed variables (continuous and discrete) of SVM mixed variables has also to be researched [22].

In a bid to overcome the limitation of working with discrete values, an algorithm that can handle mixed (discrete and continuous) values with the ability to perform the simultaneous optimization process for both feature subset selection and tuning SVM parameters has to be proposed to enhanced the classification accuracy.

3. PRELIMINARIES

In this section, basic concepts of SVM and ACO are introduced.

3.1. Support Vector Machine

In binary classification problem, there are *m* input data expressed by $T = \{(x_1, y_1), \ldots, (x_m, y_m)\}$. Let x_i denote the i^{th} example and $y_i \in \{1, -1\}$ represents the class of input data to which the i^{th} input data belong. Classifying variables $w \in R^d$ and $b \in R$ require to prove that $y_i(w^T x_i + b) \geq 1$. The hyper-plane represented by $w^T x + b = 0$ falls in the middle way between the limiting of hyper-planes given by $w^T x + b = -1$. The margin of separable through the two classes is presented by $\frac{2}{\|w\|_2}$, where $\|w\|_2$ denotes the L_2 norm of w. Assume the vectors are those training input data falling into the above two hyper planes. The classical SVMs, solutions are found by solving the following optimization problem:

$$\min_{w,b} \frac{1}{2} w^T w \quad \text{s.t.} \quad \forall i: y_i(w^T x_i + b) \geq 1 \tag{1}$$

The classification function is $f(x) = \text{sign}(w^T x + b)$, the sign function will be positive if the argument is non-negative and will be negative if the argument is negative. If two classes are not linearly separable, the classifier variables w and b are required to fulfil $y_i(w^T x_i + b) \geq 1 - \xi_i$. Figure 1 illustrates this.

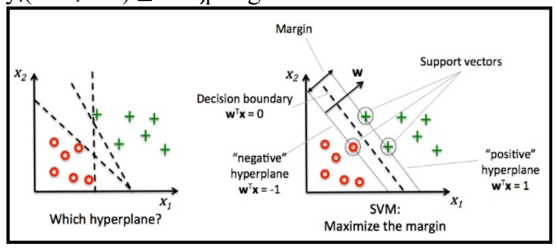

Figure 1. Support vector machine

The optimization problem of Eq. (1) may be modified to:

$$\min_{w,b} \frac{1}{2} w^T w + c \sum_{i=1}^{m} \xi_i \quad \text{s.t.} \quad \forall i: y_i(w^T x_i + b) \geq 1 - \xi_i, \xi_i \geq 0 \tag{2}$$

where c is a penalty variable and ξ_i are slack parameters . The twin optimization problem of Eq. (2) may be expresses as:

$$\min_{a} \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y_i\, y_j (x_i . x_j) á_i á_j - \sum_{i=1}^{m} á_i \quad \text{s.t.} \quad \sum_{i=1}^{m} y_i á_i = 0, 0 \leq á_i \leq c\,, i = 1, \ldots, m \tag{3}$$

where $á_i$ are Lagrangian multipliers. The optimal solution is:

$$w = \sum_{i=1}^{m} á_i^* y_i x_i, \quad b = \frac{1}{N_{sv}} \left(y_i - \sum_{i=1}^{N_{sv}} á_i^* y_i (x_i . x_j) \right) \tag{4}$$

where $á_i^*$ is the solution of the twin optimization problem Eq. (3), and N_{sv} denotes the number of support vectors fulfil $0 < \alpha < c$. The solving function is $f(x) = \text{sign}(w^T x + b)$ [37, 38].

3.2. Ant Colony Optimization

The basic properties of the ACO approach involve: (1) the utilization of a colony of ants that maximizing the strong point of the community-based approach), (2) the cooperative collaboration among the ants can powerfully solve a problem (i.e., it is a multi-agent approach), (3) greedy and stochastic nature of the approach that maximize local and universe seek capabilities, (4) the reinforcement learning approach, (5) distributed calculation because of the inherent parallelism [39]. Ant system, the initial ACO variant has the basic property that focus on the pheromone values. These values are modified in every m ants that have constructed a solution in the loop. The pheromone T_{ij}, is linked with the path connecting centre i and data points j, is modified as follows [12, 40]:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$\hat{o}_{ij} \leftarrow (1-\tilde{n}).\hat{o}_{ij} + \sum_{k=1}^{m} \Delta\hat{o}_{ij}^{k} \quad (5)$$

where ñ is the evaporation ratio , m is the count of data points, while $\Delta\hat{o}_{ij}^{k}$ is the amount of pheromone put on path (i, j) by ant k.

$$\Delta\hat{o}_{ij}^{k} = \begin{cases} \frac{Q}{L_k} & if\ ant\ k\ used\ path\ (i,j)\ in\ its\ journey \\ 0 & else \end{cases} \quad (6)$$

where Q is a fixed number and L_k is the length of the path built through ant k.

In building of a solution, ant chooses the next path to be visited by a stochastic technique. When ant 'k' is in input data example 'j' the probability of walking to centre 'i' is calculated using the equation:

$$p_{ij}^{k}(t) = \frac{[\hat{o}_{ij}(t)^{\acute{a}}].[\varsigma_{ij}]^{\hat{a}}}{\sum_{l\in J_i^k}[\hat{o}_{il}(t)^{\acute{a}}].[\varsigma_{ij}]^{\hat{a}}} \quad (7)$$

where $\hat{o}_{ij}(t)$ = quantity of pheromone linked on way (i, j), ς_{ij} = Heuristic attractiveness of selecting centre i when the ant is at input data example j, J_i^k = it is the group of input data example's unvisited centre.

The variables α and β monitor the associative necessary of the pheromone opposite the heuristic data ς_{ij}, (eta matrix) which is presented through:

$$\varsigma_{ij} = \frac{1}{d_{ij}} \quad (8)$$

where d_{ij} represent weighted Euclidean distance through center i and the input data example j:

$$d_{ij} = \|x_{jm} - v_{in}\|^2 \quad (9)$$

where n = number of centers, m = No. of input data examples.

However, due to the limitation of search mechanism, ACO is not good at dealing with mixed variables. Over the past year, a lot of attempts have been made to fill this gap. Now, based on the great development, ACO can be widely applied in mixed variables optimization decision, which can improve intelligent systems in term of data clustering, training neural network, etc. The framework of ACO consists of three parts: ant based solution construction, pheromone update, daemon action (optional). The parts are either ant-related approaches or extension of ACO to continuous functions. The first variant of ACO to solve continuous problem is ACO_R where a probability distributed function is used to generate solutions which are kept in an archive. The pheromone update is accomplished by substituting bad solutions in the archive with good solutions. Similar to ACO_R, IACOR-LS which is another variant of ACO, uses three types of local search procedures and increased the size of archive over time. ACO_{MV} divides the archive into three parts to store continuous parameters, ordinal parameters and categorical parameters and made innovation in the calculation of the weight. Thus, it can deal with not only continuous optimization but also mixed-variable optimization [41].

4. RESEARCH METHODOLOGY

The research methodology starts with the datasets development. The second phase deals with developing an intelligent classification algorithm to simultaneously optimize SVM parameter and attributes subset selection. Figure 2 depicts the phases of the research methodology.

4.1. Dataset development

Dataset development consist four steps; these steps are: (1) dataset description, (2) dataset cleaning, (3) dataset transformation, and finally (4) dataset scaling. The details of these steps are explained in below.

4.1.1. Dataset description

A collection of ten datasets from a University of California, Irvine (UCI) repository [42] have been used in this research. The datasets are Australian, Pima-Indian Diabetes, Heart (Statlog), Ionosphere, German, Sonar, Splice, Image Segmentation, Iris, and Vehicle. Table 2 summarizes the main characteristics for these datasets.

Table 2. Summary of UCI datasets

Datasets	No. of Instance	No. of Feature	No. of Class	Feature Type
Australian	690	14	2	Integer, Categorical, Real
German	1000	20	2	Integer, Categorical,
Heart (Statlog)	270	13	2	Real, Categorical
Image Segmentation	2310	19	7	Real
Ionosphere	351	34	2	Real, Integer
Iris	150	4	3	Real
Pima-Indian Diabetes	768	8	2	Real, Integer
Sonar	208	60	2	Real
Splice	3190	61	3	Categorical
Vehicle	846	18	4	Integer

4.1.2. Data cleaning

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Cleaning of the data is needed to enhance the quality of the raw data for classification which will maximize the classification accuracy. Data from real-world resources are usually incorrect, imperfect, and conflicting, possibly because of execution mistakes or system application errors. Table 3 summarizes the main characteristics for the used datasets after cleaning.

Table 3. Summary of cleaned datasets

Datasets	No. of Instances	No. of Features	No. of Classes
Australian	397	11	2
German	1000	20	2
Heart (Statlog)	270	13	2
Image Segmentation	2310	18	7
Ionosphere	351	34	2
Iris	150	4	3
Pima-Indian Diabetes	395	8	2
Sonar	208	60	2
Splice	1000	61	3
Vehicle	846	18	4

4.1.3. Dataset transformation

The Australian Credit Approval, Heart, German Credit, and Splice datasets contain categorical values. In order to deal with these datasets any categorical value were converted to numerical value. The class labels of all binary classes' datasets were also converted from 1 and 0 to +1 and -1. In the case of multi class datasets the same strategy was used, but after grouping each two classes to become binary class datasets.

4.1.4. Dataset scaling

All the datasets were scaled through the dataset development step to prevent attributes with high numerical values from dominating those in lower numerical values and to decrease the calculation efforts. All attributes were linearly scaled to [0, 1] rang using the formula [22, 25-32]:

$$\bar{x} = \frac{x - min_i}{max_i - min_i} \quad (10)$$

where x is the input attribute value, $\bar{x}$ is the scaled attribute value, and max_i and min_i are the maximum and minimum attribute values of attribute i respectively.

4.2. FORMULATION OF INTELLIGENT CLASSIFICATION ALGORITHM TO SIMULTANEOUSLY OPTIMIZE SVM PARAMETER AND ATTRIBUTE SUBSET SELECTION

The SVM's problems, which is related to selecting suitable feature subset with a few count of attributes besides the SVM variables problem, will be dealt with using mixed variables ACO variants which are mixed-variable ACO (ACO_{MV}) and Incremental mixed-variable ACO ($IACO_{MV}$). Features are represented as discrete graph nodes while C and γ SVM parameters are continuous values, so there will be a need to use ACO_{MV} that can deal with discrete, continuous, or both values' types. The number of chosen features varies from ant to ant. Hence, it is not necessary for an ant to visit all the features. The termination criterion for ant to stop its visits to the feature is when the ant arrives at a predefined selection of features which will be generated randomly. The ant's solution will represent a mix of SVM parameters which are penalty parameters C and ã for RBF kernel function and feature subset. Based on the solution archive, pheromone table and suitable features, the probability of transition is calculated to select a solution track for the ant. The pheromone table and solution archives shall be modified founded on the classification precision and attribute quality. Wrapper approach strategy, to select feature subset, will be used to hybridize ACO variant with SVM. This will use the overall classification accuracy produced through the SVM classifier and the necessary features to hybridize together into ACO variants' algorithms. In this case, the optimal selected features will be dependent on the inductive and figurative preferences of the learning algorithms that are utilized to build the SVM classifier.

5. THE PROPOSED ALGORITHMS

The proposed algorithms are ACO_{MV}-SVM and $IACO_{MV}$-SVM which are based on the mixed-variable ant colony optimization (ACO_{MV}) [43] and Liao et al.'s [44] suggestion of incremental mixed-variable ant colony optimization ($IACO_{MV}$). One of the new ACO research directions is to optimize mixed-variable (continuous and discrete) problems. ACO_{MV} is considered as the first algorithm that can handle the mixed-variable which follows the same ACO framework, while $IACO_{MV}$ suggested by Liao et al. [44] is to improve ACO_{MV} performance in solving stagnation. The proposed algorithms will execute the continuous parameter optimization and

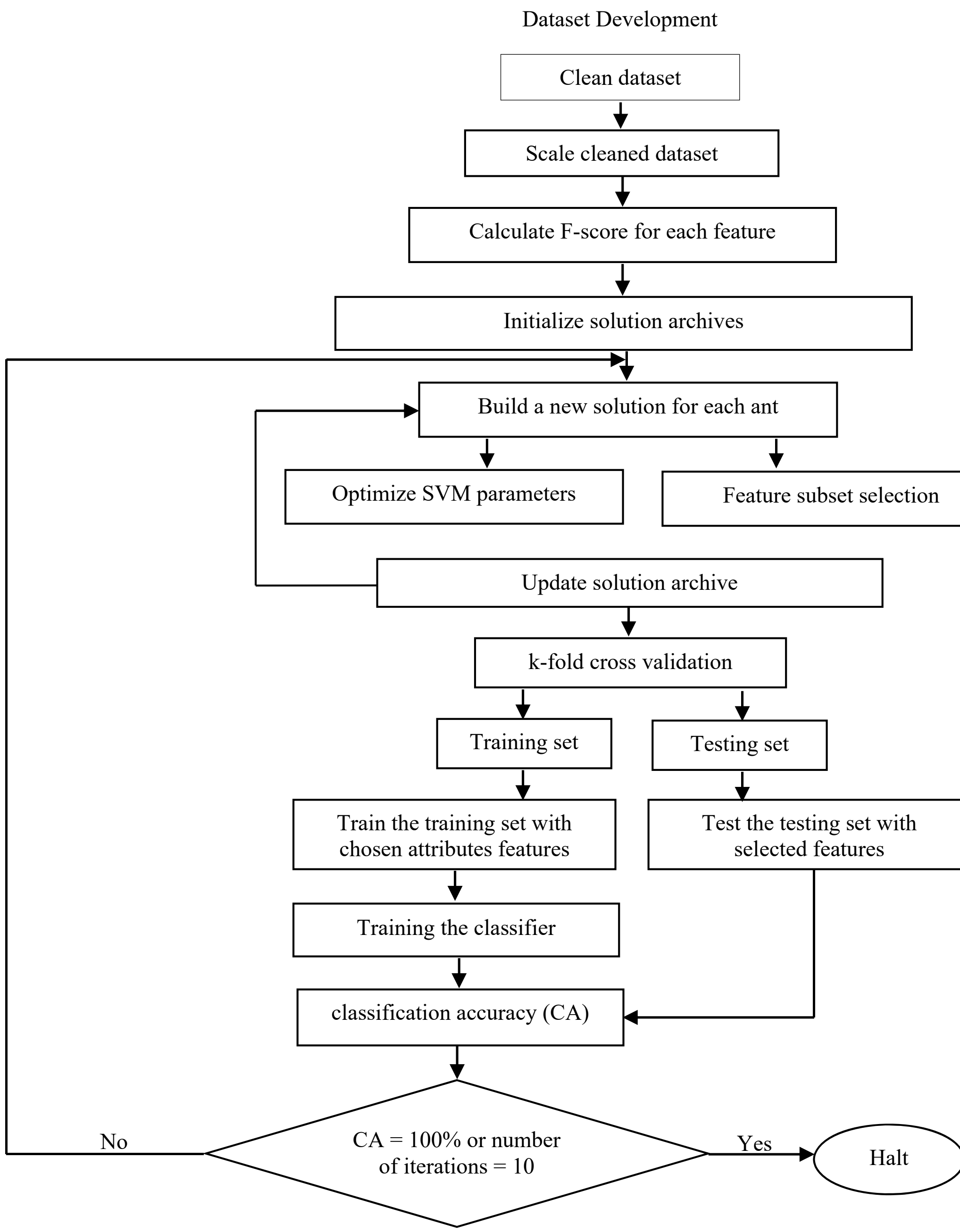

Figure 2. Algorithm's generic flow chart

SVM feature selection processes in parallel. Figure 2 presents the generic flow chart of the proposed algorithms. In the literature, algorithms in [45-48] run the two processes in sequence and with different methods of parameter optimization and feature subset selection. The difference between ACO_{MV}-SVM and $IACO_{MV}$-SVM is the size of the solution archive. The size of the archive in ACO_{MV} is fixed while in $IACO_{MV}$, the size of solution archive increases as the optimization procedure progress. The optimization procedure starts with a non-big solution archive size and a new solution is appended to the solution archive in every growth loops until a full solution archive size is achieved. Each time a new solution is append to the solution achieve, the solution achieve is initialized utilizing information taken from other solutions in the solution archive.

In the proposed solution, ACO_{MV}-SVM algorithm will tune SVM parameters using continuous ACO (ACO_R), while $IACO_{MV}$-SVM algorithm will tune SVM parameters using Incremental ACO_R ($IACO_R$). Class assignment will be done by SVM. The classification accuracy is then computed and feed to ACO algorithms to update the solution archives and pheromone tables in cases where the produced classification accuracy is not satisfied. Otherwise, if the classification accuracy is acceptable, the proposed algorithms will terminate. ACO_{MV} keeps the generated solutions' values and its fitness function values in solution archives rather than pheromone tables. This algorithm begins by initializing this solution's archives with arbitrary solutions and then ordered them according to its fitness function to influence the seek procedure in the direction of the best solutions obtained through seeking. The solutions in the solution archives are stored and ranked based on their objective function.

The proposed algorithms start by initializing three solution archives (C values, γ values, feature subset). The initialization for C and γ values will be randomly assigned by dividing the range of C and γ with variable k which represents the size of the solution archives. The solution archive for the feature subset will be initialized by storing and ranking the features according to their F-score. After initializing the solution archives, each solution will be used by the ant to build new solution. Each ant will start to construct its solution by calling the functions $ACO_{MV\text{-tune SVM parameter}}$ and $ACO_{MV\text{-feature subset selection}}$ for parameter optimization and feature selection respectively. $ACO_{MV\text{-tune SVM parameter}}$ will optimize the continuous value of SVM parameters through the use of ACO_R or $IACO_R$. Figures 3 and 4 illustrate ACO_R and $IACO_R$ respectively.

ACO_R Algorithm
Begin
Initialize k solutions
Get k solutions
//ordered the solutions and keep them in the archives
T = Order (S_1, ..., S_k)
while halting conditions is not fulfil **do**
  //create m new solutions
  for i = 1 to m **do**
    //build solution
    select S based on its weight
    sample chosen S
    keep newly created solutions
    Evaluate newly created solutions
  end
  //keep solutions and choose the good k
  T = Good (Sort S_1, ... $S_k + m$), k)
end
End

Figure 3. Pseudo code for ACO_R Algorithm

IACO Algorithm
k = *InitArhiveSize*
Adjust k solutions
Get k solutions
while halting conditions are not fulfil **do**
  // Created new solutions
  if *rand (0,1)* < p **then**
    for i = 1 to no. of ants **do**
      choose good solution
      divide good chosen solution
      evaluate the new created solution
        if Newly created solution is better than S_{best} **then**
          replace newly created solution for S_{best}
        end
    end
  else
    for j = 1 to k **do**
      Use probability to select S
      sample selected S
      store newly generated solutions
      evaluate the new generated solutions
      if newly created solution is better than S_j **then**
        replace newly created solution for S_j
      end
    end
  end
  // Archive Growth
  if present loops are multiple of *Growth* & k < *MaxArchiveSize* **then**
    reset new solution
    add new solution to the archive
    k++
  end
  //Reset Technique
  if # (number) of loops without enhancing classification accuracy of S_{best} = *MaxStagIter* **then**
    re-initialize T (solution archive) but preserving S_{best}
  end
End

Figure 4. Pseudo code of $IACO_R$ algorithm

C and γ values are calculated based on probability distributed function as shown below:

$$w_l = \frac{1}{qk\sqrt{2\eth}} e^{-\frac{(l-1)^2}{2q^2k^2}} \quad (11)$$

and

$$p_l = \frac{w_l}{\sum_{r=1}^{k} w_r} \quad (12)$$

where l is the index of $solution_l$ in the solution archive and k is the count of attributes.

The constructed solution will be sent to the SVM together with the feature subset generated from the second part of $ACO_{MV\text{-feature subset selection}}$ in ACO_{MV}-SVM. Based on the outcome of SVM, the solution archives will update depended on SVM classification accuracy.

$ACO_{MV\text{-feature subset selection}}$ algorithm that is used to construct feature subset is shown in Figure 5.

Input: Features
Output: Optimal feature subset
Begin
  calculate feature subset size randomly
  initialize pheromone table
  for i = 1 to no. of features **do**
    //select first feature in feature subset
    compute weight for each feature
    compute probability for each feature
    select feature with highest probability
    append feature with highest probability to feature subset
    remove appended feature from original features set
  end
    for j = 1 to feature subset size - 1 **do**
      //select other features in feature subset
      compute probability for remaining features
      select feature with highest probability
      append feature with highest probability to feature subset
      remove appended feature from original features set
    end
    update pheromone table
End

Figure 5. Pseudo code of proposed enhanced algorithm for feature subset selection

The $ACO_{MV\text{-feature subset selection}}$ algorithm starts by computing the size of the attribute subset for each ant and then initialized a solution archive using the following formula:

$$phero_{feature_i feature_j} = \frac{1}{\sum_{i=1}^{no.of\ feature} F\text{-}score_{feature}} \quad (13)$$

The ant will then start to construct its feature subset. The first feature in the feature subset will be selected according to its probability calculated as follows:

$$P = \frac{w_{feature_i} * F\text{-}Score_{feature_i}}{\sum_{i=1}^{no.of\ features} w_{feature_i}} \quad (14)$$

where the $F-Score_{feature_i}$ is computed according to Fisher formula and $w_{feature_i}$ is the weight of $feature_i$ and it is computed as

$$w_{feature_i} = \frac{w}{u} + \frac{q}{ç} \quad (15)$$

where w is computed using Eq. (11), u as a counter that counts how many times $feature_i$ has been selected, q is the algorithm's variable to monitor the diversification of seek procedure, and ç is the number of unselected features.

The reason for using the probability function instead of using the standard established discrete probability is because there is a need to traverse from continuous variables (SVM parameters) to discrete (feature subset) variables. After selecting the first feature in the feature subset, the ant continues to build its feature subset by selecting other features and appending them to the feature subset. The selection of other features is completed through computing the probability for each of the features as follows:

$$Prob_{ij}^{k}(t) = \begin{cases} \frac{(phero_{feature_i feature_j})^{â}(F\text{-}Score_{feature_i})^{â}}{\sum_{j \in I_i^k}(phero_{feature_i feature_j})^{â}(F\text{-}Score_{feature_i})^{â}} & if\ j \in I_i^k \\ 0 & otherwise \end{cases} \quad (16)$$

where $phero_{feature_i feature_j}$ is the pheromone value on the arc that connects $feature_i$ and $feature_j$. Fisher score is equal to $F-Score_{feature_i}$. After all ants have finished building the feature subset, pheromone will be updated using the following formula:

$$phero_{feature_i feature_j}(t+1) = p * phero_{feature_i feature_j} + \sum_{k=1}^{no.of\ ants} \Delta phero_{feature_i feature_j}{}^{k}(t) \quad (17)$$

where p is an arbitrary number generated in the range of (0, 1), $phero_{feature_i feature_j}$ is the current pheromone on the edge that connects $feature_i$ and $feature_j$, and $\Delta phero_{feature_i feature_j}{}^{k}$ is computed as:

$$\Delta phero_{feature_i feature_j}{}^{k} = \begin{cases} CVACC^{K} * weight_{feature_i}^{k} * weight_{feature_j}^{k} & if\ ant\ k\ use\ edge \\ 0 & else \end{cases} \quad (18)$$

$CVACC^k$ is the cross validation classification accuracy generated by SVM from ant_k solution, $weight_{feature_i}^{k}$ and $weight_{feature_j}^{k}$ are the weights of $feature_i$ and $feature_j$ respectively generated from

SVM on the training set from ant_k solution. Ultimately, ant solution will involve three parts; the first two parts is related to SVM parameters which include the C and γ values, and the third part is related to attribute subset.

6. EXPERIMENTAL RESULT

The proposed algorithms were tested on eight datasets from the UCI repository.

C programming language was utilized to program the proposed algorithms. Experiments were run on an Intel(R) Core (TM) 2 duo CPU T5750, running at 2.00 GH_Z with 4.00 GB RAM and 32-bit operating system.

k-fold cross validation technique is one of the most popular resampling procedures to evaluate algorithms on a limited data sample. This procedure has also been used by [49] in multi-label classification and by [50] in building multiple classifier system using ant system. Thus, the *k*-fold cross-validation procedure was applied in the process of obtaining the classification accuracy. A set of labelled samples are randomly partitioned into k disjoint folds of equal size. Then, one of the k folds is randomly selected as the testing set and the remaining (k-1) folds are selected as the training set with the assumption that there is at least one sample per class.

Classification accuracy has been used in evaluating the performance of the proposed algorithms with other common algorithms. The classification accuracy (acc) is the ratio of numbers of all correctly classified instances and the total number of instances as shown in Equation 19.

$$acc = \frac{no.of\ all\ correctly\ classified\ instances}{total\ number\ of\ instance} \times 100 \quad (19)$$

The estimation of classification accuracy is obtained by dividing the total of all classification accuracies by the total number of folds or rounds as shown in Equation 20.

$$acc_{CV} = \frac{1}{k}\sum_{i=1}^{k} acc_i \quad (20)$$

acc_i is the classification accuracy of round i and k is the number of folds.

In these experiments, the search range for C was [2^{-1}, 2^{12}] and γ [2-12, 22]. The number of ants is 2 and q value is 0. $\alpha = 1$ and $\beta = 2$ are the recommended values according to [51]. Finally the *Growth* test values were 2, 4, 6, 8, and 10. The results show that the best value for *Growth* was 5. For *Stag* test, the values that were used are 1, 2, 3, 4, and 5 and results show that the best value is 2. The Initial solution archive size test values were 2, 4, 6, 8, 10, 12, and 14. The results show that the best value for the initial solution was 10, while for maximum solution archive size test the values were 3, 5, 7, 9, 11, 13, and 15 and the results show that the best value was 15.

The behaviour of the proposed algorithms is assessed through comparison with Gravitational Search Algorithm (GSA)-SVM [22], $GA_{with\ feature\ chromosome}$-SVM [25], Ant Colony Optimization (ACO)-SVM [26], CSO-SVM [30], Clonal Selection Algorithm (CSA)-SVM [27], Particle Swarm Optimization (PSO)-SVM [33], Simulated Annealing (SA)-SVM, and GA-SVM [32]. Table 4 summarizes the performance statistics for ACO_{MV}-SVM, $IACO_{MV}$-SVM, GSA-SVM, $GA_{with\ feature\ chromosome}$-SVM, ACO-SVM, CSO-SVM, CSA-SVM, PSO-SVM, SA-SVM, and GA-SVM.

Table 4. Comparison of classification accuracy

Datasets	1	2	3	4
Australian	96.33 ± 0.91	**96.96 ± 0.53**	91.59 ± 2.14	90.82
German	96.16 ± 0.57	**97.23 ± 0.46**	86.10 ± 1.97	86.40
Heart	97.70 ± 0.93	**98.01 ± 0.35**	95.56 ± 2.34	92.59
Ionosphere	99.86 ± 0.25	**99.99 ± 0.02**	99.43 ± 1.21	98.56
Iris	99.95 ± 0.08	99.98 ± 0.03	**100 ± 0**	**100**
Pima-India Diabetes	95.07 ± 1.73	**97.22 ± 0.81**	83.84 ± 5.14	82.70
Sonar	99.94 ± 0.97	**99.99 ± 0.02**	99.00 ± 2.11	98.80
Vehicle	93.32 ± 2.20	**93.92 ± 0.29**	88.24 ± 1.47	90.20
Datasets	**5**	**6**	**7**	**8**
Australian	93.77 ± 2.27	91.03	92.19 ± 3.23	88.10 ± 2.25
German	82.20 ± 2.82	81.62	-	85.60 ± 1.96
Heart	97.04 ± 2.34	-	-	94.80

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

				± 3.32
Ionosphere	99.43 ± 1.2	99.01	99.07 ± 0.73	98.56 ± 2.03
Iris	99.33 ± 2.1	99.20	-	**100 ± 0**
Pima-India Diabetes	84.73 ± 5.37	82.68	82.22 ± 3.55	81.50 ± 7.13
Sonar	98.10 ±3.33	96.26	95.99 ± 3.90	98.00 ± 3.5
Vehicle	90.77 ± 2.7	89.83	90.14 ± 2.21	84.06 ± 3.54

1 Proposed ACO_{MV}-SVM Algorithm
2 Proposed $IACO_{MV}$-SVM Algorithm
3 $GA_{\text{with feature chromosome}}$-SVM
4 CSA-SVM 5 CSO-SVM 6 PSO-SVM
7 SA-SVM 8 GA-SVM

Highest classification accuracy was obtained by $IACO_{MV}$-SVM algorithm in all datasets except for the Iris dataset where GA $_{\text{with feature Chromosome}}$-SVM, CSA-SVM, while GA-SVM obtained perfect results. On the other hand, the proposed ACO_{MV}-SVM algorithm performed second best on all datasets except for the Iris dataset. Good performances by the proposed algorithms were observed because of the simultaneous process of tuning the SVM variables and selecting the feature subset which eliminate the accumulation of error from one process to another. Error is eliminated when continuous value was not discretized.

In this research, ACO variants have been hybridized with SVM through a wrapper-based feature selection approach. The approach has provided the ability of change the feature subset until good classification accuracy is obtained. This is also due to the fact that feature subset selection via the wrapper approach is dependent on the inductive learning approach. The kernel function that has been used in this study is the RBF because it requires small number of parameters and has been proven to produce good results in many problems. A binary SVM classifier as well as a multi-class SVM classifier through utilizing One-Against-One has also been used.

7. CONCLUSION

ACO_{MV} and $IACO_{MV}$ as extensions of the ACO algorithm present the opportunity to manage mixed-variable (discrete and continuous) optimization problems SVM. Results of using the proposed algorithms are encouraging as compared to other common algorithms. There are many directions to improve this work. Some of these directions are: the proposed algorithms can be used with Support Vector Regression which has the same problem as SVM as pointed out by [26, 29]. This work needs several modifications to the proposed algorithms. Another suggestion is to utilize the proposed algorithms in tackling dynamic problems. In dynamic optimization problems, the seek domain changes with time. Other types of SVM like least square SVM may be utilized to tackle classification problems. Other kernel function beside Radial Basis Function may also be experimented. Building a technique for utilizing different kernel functions and selecting the most prosperous kernel function that will provide the best classification result can be another future work. Other benchmark datasets and real world problems can be used to prove the success of the proposed algorithms.

ACKNOWLEDGMENT

This work was supported by the Long Term Research Grant Scheme, S/O code 12490.

REFERENCES

[1] J. Zhang, S. O. Williams and H. Wang, "Intelligent computing system based on pattern recognition and data mining algorithms," *Sustainable Computing: Informatics and Systems,* vol. 20, p. 192-202, December 2018.
[2] N. K. Sreeja and A. Sankar, "Pattern matching based classification using ant colony optimization based feature selection," *Applied Soft Computing*, vol. 31, p. 91-102, June 2015.
[3] D. Kumar and M. Thakur, "All-in-one multicategory least squares nonparallel hyperplanes support vector machine," *Pattern Recognition Letters,* vol. 105, p. 165-174, April 2018.
[4] P. H. Kassani, A. B. J. Teoh and E. Kim, "Evolutionary-modified fuzzy nearest-neighbour rule for pattern classification," *Expert Systems with Applications,* vol. 88, p. 258-269, December 2017.
[5] F.-J. Gonzalez-Serrano, A. Navia-Vazquez and A. Amor-Martin, "Training support vector machine privacy-protected data," *Pattern Recognition,* vol. 72, p. 93-107, December 2017.
[6] K. Gajalakshmi, S. Palanivel, N. J. Nalini and S. Saravanan, "Automatic classification of cast

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

iron grades using support vector machine," *Optik*, vol. 157, p. 724-732, March 2018.

[7] D. Liu, Y. Shi and Y. Tian, "Ramp loss nonparallel support vector machine for pattern classification," *Knowledge-Based Systems,* vol. 85, p. 224-233, September 2015.

[8] X. Zhang, S. Ding and Y. Xue, "An improved multiple birth support vector machine for pattern classification," *Neurocomputing*, vol. 225, p. 119-128, February 2017.

[9] M. Wang, Y. Wan, Z. Ye and X. Lai, "Remote sensing image classification based on the optimal support vector machine and modified binary coded ant colony optimization algorithm," *Information Sciences*, vol. 402, p. 50-68, September 2017.

[10] Y. Zhou, F. He, N. Hou and Y. Qiu, "Parallel ant colony optimization on multi-core SIMD CPUs," *Future Generation Computer Systems,* vol. 79, p. 473-487, February 2018.

[11] P. Shunmugapriya, and S. Kanmani, "A hybrid algorithm using ant and bee colony optimization for feature selection and classification (AC-ABC Hybrid)," *Swarm and Evolutionary Computation,* vol. 36, p. 27-36, October 2017.

[12] B. Biswal, P. K. Dash and S. Mishra, "A hybrid ant colony optimization technique for power signal pattern classification," *Expert System with Applications,* vol. 38, p. 6368-6375, May 2011.

[13] H. J. A. Nasir, K. R. Ku-Mahamud and A. M. Din, "Load balancing using enhanced ant algorithm in grid computing," *Proceedings of The 2nd International Conference on Computational Intelligence, Modelling and Simulation*, Bali, Indonesia, September 28-30, 2010, pp. 160-165.

[14] H. J. A. Nasir and K. R. Ku-Mahamud, "Grid load balancing using ant colony optimization," *Proceeding of The* 2nd *International Conference on Computer and Network Technology,* Bangkok, Thailand, April 23-25, 2010, pp. 207-211.

[15] Y. Zhang, W. S. Lee, M. Li, L. Zheng and M. A. Ritenour, "Non-destructive recognition and classification of citrus fruit blemishes based on ant colony optimized spectral information," *Postharvest Biology and Technology,* vol. 143, p. 119-128, September 2018.

[16] F. Abbas and P. Fan, P., "Clustering-based reliable low-latency routing scheme using ACO method for vehicular networks," *Vehicular Communications,* vol. 12, p. 66-74, April 2018.

[17] G. N. Reddy and S. P. Kumar, "Modified ant colony optimization algorithm for task scheduling in cloud computing systems," *Smart Innovation, Systems and Technologies,* vol. 104, p. 357-365, January 2019.

[18] H. Ghimatgar, K. Kazemi, M. S. Helfroush and A. Aarabi, "An improved feature selection algorithm based on graph clustering and ant colony optimization," *Knowledge-Based Systems,* vol. 159, p. 270-285, November 2018.

[19] A. Husin and K. R. Ku-Mahamud, "Ant system and weighted voting method for multiple classifier system," *International Journal of Electrical and Computer Engineering*, *vol.* 8, p. 4705-4717, December 2018.

[20] H. B. Alwan and K. R. Ku-Mahamud, "Mixed-variable ant colony optimisation algorithm for feature subset selection and tuning support vector machine parameter," *International Journal of Bio-Inspired Computation,* vol. 9, p. 53-63, 2017.

[21] M. Aladeemy, S. Tutun, and M. T. Khasawneh, "A new hybrid approach for feature selection and support vector machine model selection based on self-adaptive cohort intelligence," *Expert Systems with Applications*, vol. 88, p. 118–131, December 2017.

[22] S. Sarafrazi and H. Nezamabadi-pour, "Facing the classification of binary problems with a GSA-SVM hybrid system," *Mathematical and Computer Modelling,* vol. 57, p. 270–278, January 2013.

[23] M. Zavar, S. Rahati, M.-R. Akbarzadeh and H. Ghasemifard, "Evolutionary model selection in a wavelet-based Support Vector Machine for automated seizure detection," *Expert Systems with Applications*, vol. 38, p. 10751–10758, September 2011.

[24] H. Han, W. Hou-jun and D. Xiucheng, "Transformer fault dignosis based on feature selection and parameter optimization," *Energy Procedia*, vol. 12, p. 662–668, 2011.

[25] M. Zhao, C. Fu, L. Ji, K. Tang and M. Zhou, "Feature selection and parameter optimization for support vector machines: A new approach based on genetic algorithm with feature chromosomes," *Expert System with Applications,* vol. 38, p. 5197–5204, May 2011.

[26] C.-L. Huang, "ACO-based hybrid classification system with feature subset selection and model parameters optimization," *Neurocomputing*, vol. 73, p. 438–448, December 2009.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[27] S. Ding and S. Li, "Clonal selection algorithm for feature selection and parameters optimization of support vector machines," *Proceedings of the 2nd International Symposium on Knowledge Acquisition and Modelling*, Wuhan, China, November 30–1 December, 2009, pp. 17–20.
[28] C.-L. Huang and J. Dun, "A distributed PSO-SVM hybrid system with feature selection and parameter optimization," *Applied Soft Computing,* vol. 8, p. 1381–1391, September 2008.
[29] C. Huang and C. Wang, "A GA-based feature selection and parameters optimization for support vector machines," *Expert Systems with Applications,* vol. 31, p. 231–240, August 2006.
[30] K. Lin and H. Chien, "CSO-based feature selection and parameter optimization for support vector machine," *Proceedings of The Joint Conference on Pervasive Computing*, December 3–5, 2009, Tamsui, Taipei, China, pp. 783–788.
[31] D. T. Pham, Z. Muhamad, M. Mahmuddin, A. Ghanbarzadeh, E. Koc and S. Otri, "Using the bees algorithm to optimise a support vector machine for wood defect classification," *In: Innovative Production Machines and Systems Virtual Conferences (IPROMS)*, Cardiff, UK, 2007.
[32] S.-W. Lin, Z.-J. Lee, S.-C. Chen and T.-Y. Tseng, "Parameter determination of support vector machine and feature selection using simulated annealing approach," *Applied Soft Computing,* vol. 8, p. 1505–1512, September 2008.
[33] S.-W. Lin, K.-C. Ying, S.-C. Chen and Z.-J. Lee, "Particle swarm optimization for parameter determination and feature selection of support vector machines," *Expert Systems with Applications,* vol. 35, p. 1817–1824, November 2008.
[34] S. M. Vieira, L. F. Mendonça, G. J. Farinha and J. M. C. Sousa, "Modified binary PSO for feature selection using SVM applied to mortality prediction of septic patients," *Applied Soft Computing,* vol. 13, p. 3494–3504, August 2013.
[35] X. Zhang, and L. Jiao, "Simultaneous feature selection and parameters optimization for SVM by Immune Clonal Algorithm," In: Wang L., Chen K., Ong Y.S. (eds) *Advances in Natural Computation. ICNC 2005. Lecture Notes in Computer Science,* vol. 3611. Springer, Berlin, Heidelberg.
[36] S. Maldonado, R. Weber and J. Basak, "Simultaneous feature selection and classification using kernel-penalized support vector machines," *Information Sciences*, vol. 181, p. 115–128, January 2011.
[37] X. Xie and S. Sun, "PAC-Bayes bounds for twin vector machines," *Neurocomputing,* vol. 234, p. 137-143, April 2017.
[38] Y. Wan, S. Song, G. Huang and S. Li, "Twin extreme learning machines for pattern classification," *Neurocomputing,* vol. 260, p. 235-244, October 2017.
[39] S. Tabakhi, A. Najafi, R. Ranjbar and P. Moradi, "Gene selection for microarray data classification using a novel ant colony optimization," *Neurocomputing,* vol. 168, p. 1024-1036, November 2015.
[40] A. Byrski, E. Swiderska, J. Lasisz, M. Kisiel-Dorohinicki, T. Lenaerts, D. Samson, B. Indurkhya and A. Nowe, "Socio-cognitively inspired ant colony optimization," *Journal of Computational Science,* vol. 21, p. 397-406, July 2017.
[41] Z. Chen, S. Zhou and J. Luo, "A robust ant colony optimization for continuous functions," *Expert Systems with Applications,* vol. 81, p. 309-320, September 2017.
[42] *UCI Repository of Machine Learning Databases, Department of Information and Computer Science, University of California, Irvine, CA, <http://www.ics.uci.edu/mlearn/MLRepository>, 2012.*
[43] M. Dorigo and T. Stützle, *Ant Colony Optimization.* Cambridge, England: MIT Press, 2004.
[44] K. Socha, "Ant colony optimization for continuous and mixed-variables domain," Ph.D. dissertation, Universite' Libre de Bruxelles, 2008.
[45] T. Liao, "Improved ant colony optimization algorithms for continuous and mixed discrete-continuous optimization problems," IRIDIA, Universite Libre de Bruxelles: Brussels, Belgium. No. 2010/2011, 2011.
[46] A. Garde, A. Voss, P. Caminal, S. Benito and B. F. Giraldo, "SVM-based feature selection to optimize sensitivity–specificity balance applied to weaning," *Computers in Biology and Medicine,* vol. 43, p. 533–540, March 2013.
[47] Z. Liu, H. Cao, X. Chen, Z. He and Z. Shen, "Multi-fault classification based on wavelet SVM with PSO algorithm to analyze vibration signals from rolling element bearings,"

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Neurocomputing, vol. 99, p. 399–410, January 2013.

[48] K.-C., Lin, L.-D., Huang, & Jason, C., Hung, "A novel feature selection method for support vector machines using a lion's algorithm," In: Park J., Pan Y., Kim CS., Yang Y. (eds) Future Information Technology. Lecture Notes in Electrical Engineering, vol. 309. Springer, Berlin, Heidelberg.

[49] R. Mohamed, M. N. S. Zainudin, N. Sulaiman, T. Perumal and N. Mustapha, "Multi-label classification for physical activity recognition from various accelerometer sensor positions," vol. 17, p. 209-231, April 2018.

[50] A. Husin and K. R. Ku-Mahamud, "Ant system and weighted voting method for multiple classifier systems," *International Journal of Electrical and Computer Engineering (IJECE),* vol. 8, p. 4705-4712, December 2018.

[51] Y. Fei, "Simultaneous support vector selection and parameter optimization using Support Vector Machines for sentiment classification," *Proceedings of the 7th IEEE International Conference on Software Engineering and Service Science, ICSESS 2016*, Beijing, China, August 26-28, 2016, pp. 59–62.