Journal of Computer Science

Original Research Paper

Enhanced Support Vector Machine based on Metaheuristic Search for Multiclass Problems

[1]Abdullah Saeed Ghareb*, [2,3]Mohammed Hamid Afif*, [4]Abdel-Rahman Hedar,
[5]Taysir H. Abdel Hamid and [1]Abdulgbar Saif

[1]Faculty of Information Technology and Computer Sciences, University of Sheba Region, Marib, Yemen
[2]Deptartment of Management Information Systems, College of Business Administration,
Prince Sattam Bin Abdulaziz University, Saudi Arabia
[3]Deptartment of Management Information Systems, Faculty of Administrative Science, Hadramout University, Yemen
[4]Deptartment of Computer Science, Faculty of Computers and Information, Assiut Universit, Assiut 71526, Egypt
[5]Deptartment of Information Systems, Faculty of Computers and Information, Assiut University, Assiut 71526, Egypt

Article history
Received: 20-02-2020
Revised: 07-04-2020
Accepted: 08-07-2020

Corresponding Author:
Abdullah Saeed Ghareb
Faculty of Information Technology and Computer Sciences, University of Sheba Region, Marib, Yemen
Email: aghurieb@gmail.com

Mohammed Hamid Afif
Department of Management Information Systems, College of Business Administration, Prince Sattam bin Abdulaziz University, Saudi Arabia
Email: M.afif@psau.edu.sa

Abstract: Machine learning is an important field of artificial intelligent researches and it highly growing for real intelligent applications systems that relate brain computer interface to human brain activities. Support Vector Machine (SVM) is a popular machine learning approach, which can be used for pattern recognition, prediction and classification with many diverse applications. However, the SVM has many parameters, which have significant influences on the performance of SVM in terms of its prediction accuracy that is very important measure specifically with critical applications such that used in Medical applications. This paper proposed an enhanced SVM, which employs a meta-heuristic method, called scatter search to determine the optimal or near optimal values of the SVM parameters and its kernel parameters in multi-classification problem. Scatter search has the potential to determine the appropriate values of parameters for machine learning algorithms due its flexibility and sophistication. Therefore, the proposed method integrates the advantages of scatter search method with SVM to specify the appropriate setting of SVM parameters. The experimental results on lung cancer datasets and other standard datasets prove that the scatter search is practical method for tuning SVM parameters and enhance its performance, where the achieved results are better and comparable to other related methods.

Keywords: Support Vector Machine, Meta-Heuristic Search, Scatter Search, Classification, Parameter Tuning, Lung Cancer

Introduction

Support Vector Machine (SVM) is one of the promising and effective machine learning algorithms for regression and classification problems (Tharwat, 2019), it's learning behavior depends on statistical learning theory (Cortes and Vapnik, 1995; Vapnik and Vapnik, 1998). Due to promising performance of SVM, it is applied in several domains, for example; bioinformatics (Schölkopf *et al*., 2004), text classification (Joachims, 2002), fault diagnosis (Zhang *et al*., 2015) and pattern recognition (Burges, 1998). SVM originally is designed to solve the binary classification problems. Furthermore, it is not easy to extend it for solving problems with more than two classes, known as multiclass classification problem, It is still an ongoing research area (Tharwat, 2019; Allwein *et al*., 2000). Usually, a decomposition strategy is employed for multiclass classification problem, in this case; the multiclass problem is partitioned into several binary sub-problems. Then, SVM is applied to each of these binary sub-problems to classify them; the outputs of these sub-problems are combined to get the class of new instances. Thus the automatically setting of binary and kernel parameters of SVM is not trivial task and it has major effect on performance and simplicity of SVM classification model. However, more analysis is needed for optimization to avoid the black and random selections of SVM parameters, which affect performance and create a complex prediction model (Tharwat, 2019; Zhang *et al*., 2015; Allwein *et al*., 2000). Therefore, SVM parameters must be determined carefully during training process to enhance prediction and select the accurate classification model. In other word, inappropriate parameters setting will leads to poor and irrelevant knowledge that degrades model performance (Burges, 1998; Allwein *et al*., 2000; Keerthi and Lin, 2003). In the decomposition of multiclass applications, the optimal parameter values for each of the binary SVM may differ. Generally, the empirical search for these values through a trial and error investigation method is clearly impractical. One other visible and strong solution is an investigation of the optimization techniques behaviors for SVM parameters values tuning.

This paper proposed an enhanced version of SVM based on Scatter Search (SS) optimization approach that used for automatically setting of the SVM parameters and its kernel parameters that contained in decomposition methods in multiclass prediction problem. SS approach is used due to its flexibility, since each of its elements or steps can be implemented in a variety of ways and it has acceptable degree of sophistication (Martí *et al*., 2006). Furthermore, SS can find solutions of a higher average quality earlier during the search more than some meta-heuristic search methods (Campos *et al*., 2001). Additionally, using some optimization algorithms usually loss some of relevant solutions and trap in local optimum problem which ignore or do not reaches best solutions in many cases. Thus, it is inevitable to apply one of meta-heuristic optimization strategy such as SS to find global optimum solution.

The rest of the paper is organized as follows; the next section presents reviews of related works and gives basic knowledge of SVM. Section 3 describes the stages of proposed method and section 4 reports and discusses the experiments and results for two collections of datasets. Finally, the conclusion makes up section 5.

Related Works

The SVM is an effective learning method was initially proposed to solve binary classification problem and maturated for multiclass classification of some application systems. While an inappropriate parameters, setting is a major problem of SVM that possibly result in poor and irrelevant information that degrades SVM model performance. Recently, several parameters tuning approaches for SVM have been proposed, but they have mainly been focused on trial and error investigation methods and applied to binary classification. However another solution is possible which set SVM parameters based on meta-heuristic search algorithms that locate best values (best solution) for parameters in global way. The following briefly discuss the basic concept of SVM with its parameters that require setting optimization and provide overview of several approaches of parameters tuning for SVM.

The basic idea of SVM is to implicitly map the training data into a high-dimensional feature space. A hyper-plane is constructed in this feature space, which maximizes the margin of separation between the hyper-plane and the points lying nearest to it (called the support vectors) (Tharwat, 2019; Cortes and Vapnik, 1995; Vapnik and Vapnik, 1998; Palaniswami *et al*., 2000). The hyper-plane can be used as a basis for classifying vectors of uncertain type. In a case of linearly separable data, the problem of two-category classification is stated as the follows.

Suppose that there are N training pair (x_i, y_i), where x_i is an object and y_i is a class label (±1) and $i = 1$ *to* N. The hyper-plane is defined by a discriminate function as follows:

$$f(x) = w^T \cdot x + b = 0, \quad (1)$$

where, the vector w of dimension equal to that of x and scalar b are chosen such that:

$$w^T \cdot x_i + b \geq 0 \quad if\ y_i = +1, \quad (2)$$

$$w^T \cdot x_i + b \leq 0 \quad if\ y_i = -1. \quad (3)$$

Classification of an unknown vector x into class label y(±1) is done using the discriminate function and defined as:

$$y = sign(f(x)) \quad (4)$$

In the non-separable case, the basic idea in design the nonlinear SVM is to map input vectors $x \in R^n$ into vectors $\Phi(x)$ of a higher dimensional feature space with m features (where Φ: $R^n \to R^m$). Then, solve a linear classification problem in this new feature space. To avoid an explicit representation $\Phi(x)$ of the feature space, kernel trick is applied. Kernel function $K(X, Z) = (\Phi(X) \cdot \Phi(Z))$ is a function that perform mapping from input space into higher dimension feature space. After that, a linear machine is used to classify the data in the feature space. Several kernel functions assist SVM in obtaining the optimal solution. The most frequently used as kernel functions are the Polynomial, Sigmoid, Gaussian and Radial Basis Function (RBF). The RBF and Gaussian kernels are frequently used by most studies, more details can be found in literature such as (Tharwat, 2019; Zhang *et al*., 2015; Faris *et al*., 2018; Tuba and Stanimirovic, 2017; Yin and Yin, 2016; Maglogiannis *et al*., 2009; Samadzadegan and Ferdosi, 2012; Jia *et al*., 2011; Sartakhti *et al*., 2012; Li-Xia *et al*., 2011; Chen *et al*., 2011; Lin *et al*., 2008a; 2008b; Huang and Wang, 2006; Pai and Hong, 2005a; 2006; 2005b).

One of the major problem that face SVM is how to choose the appropriate value of it is parameters, while unsuitable setting lead to classifier with poor performance (Tharwat, 2019; Zhang *et al*., 2015; Keerthi and Lin, 2003). The parameters that should be optimized are the complexity parameter C, epsilon and tolerance t and the kernel function parameters, such as σ for Gaussian kernel. The parameter C determines the trade-off between the fitting error minimization and model complexity (Zhang *et al*., 2015; Faris *et al*., 2018; Tuba and Stanimirovic, 2017; Yin and Yin, 2016; Li *et al*., 2011; Wu *et al*., 2007; Ren and Bai, 2010; Cherkassky and Ma, 2004; Liu and Jiang, 2008), where its value indicates the error expectation in the classification process of the sample data and it impacts the number of support vectors generated by the classifier (Liu and Jiang, 2008). The employment of a decomposition strategy in multiclass problems increases the number of parameter values to be determined, since every binary classifier deal with different classification problem and may have distinct ideal parameter values. Authors in (Lorena and De Carvalho, 2008) summarize three methods that can be followed to set the value of the parameters:

I. Use default values for SVM and its kernel, where each tool of SVM may define or set values for each SVM parameters and its kernel
II. Set the values manually by trial and error
III. Tune the values via some optimization techniques, such as simulated annealing, particle swarm, Genetic Algorithm (GA) and many others

In literature, some works were conducted to solve parameters tuning of SVM in multiclass decomposition. For instance, GA used in (Lorena and De Carvalho, 2008; Samadzadegan *et al*., 2010) for parameters tuning of SVM. The code matrix strategy is used in the first method for decomposition the multiclass problem. In addition, the method conducts two types of experiments: first, use different values of parameters for each binary classifier, while in the second experiments the same values of parameters are used for all binary classifiers. The authors via their experiments claim that the GA is able to get the solutions that reduce validation error rate (Lorena and De Carvalho, 2008). Since the method, proposed in (Samadzadegan *et al*., 2010) uses OAO and OAA methods for multiclass, as well as the method used the same values of the parameters for all binary classifiers. However, several methods are proposed in literature for finding the best values of SVM parameters and its kernel parameters, which focus only on problems with binary class. Many different techniques are employed like grid search (Hsu and Lin, 2002a; LaValle *et al*., 2004), GA (Pai and Hong, 2005a; 2006; Ren and Bai, 2010), Simulated Annealing (SA) (Jia *et al*., 2011; Sartakhti *et al*., 2012; Lin *et al*., 2008a; Pai and Hong, 2006; 2005b), Particle Swarm Optimization (PSO) (Lin *et al*., 2008b; Li-Xia *et al*., 2011; Ren and Bai, 2010; Sudheer *et al*., 2011; Lins *et al*., 2012) and other methods such that presented in (Zhang *et al*., 2015; Faris *et al*., 2018; Tuba and Stanimirovic, 2017; Yin and Yin, 2016; Samadzadegan and Ferdosi, 2012). Authors in (Hsu and Lin, 2002a) and (LaValle *et al*., 2004) use a grid search algorithm to find near optimal value of C and σ, when Gaussian kernel function is used. However, this method is time consuming and does not perform well. Another study (Lin *et al*., 2008a) claims that the setting search interval is a problem, where too large interval wastes computing power. On the other hand, too small interval might render a satisfactory outcome impossible. Pai and Hong (2005a) present an approach that combine GA and SVM. Their model imitates chromosome coding in their GA to generate a set of parameter values for SVM. Additionally, Wu *et al*. (2007) use a real-valued GA to optimize the parameters of SVM for predicting bankruptcy; the suggested technique is tested on the forecasting of financial crisis in Taiwan, where the presented results were promising. Researchers conclude that integrating the RGA with SVM is very successful. Pai and Hong (2006; Pai and Hong, 2005) also present SA approach to obtain parameter values for SVM and applied it to real data. In addition, a hybrid prediction method called SA-SVM proposed for predicting synthesis characteristics of the hydraulic valve; SA is used to optimize the parameters of SVM (Jia *et al*., 2011). Authors prove via experiments the strategy is applicable to forecasting the synthesis characteristics of hydraulic valve with higher accuracy rate. Lins *et al*. (2012) in propose method for reliability prediction, where PSO method used to solving the parameters setting problem of SVM. In (Faris *et al*., 2018), researchers suggested a new approach based on meta-heuristic called multi-verse optimizer for tuning the parameters of SVM, the suggested method was implemented and tested on two different system architectures and the obtained results was very promising. In (Samadzadegan and Ferdosi, 2012), authors use bees algorithm to optimize the SVM parameters as well as feature selection. Also, they compared their work with other methods likes GA and grid search, their method was the best in all performance aspects. However, all of these methods depend on prior knowledge, user expertise, or experimental trial with black box of parameters values. Hence, there is no guarantee that the parameters values obtained are optimal (Tharwat, 2019; Ren and Bai, 2010).

On the other side, SS is a population-based algorithm which was first proposed by F. Glover in the 1970's (Glover, 1977), based on some results reported in 1968. However, the SS template in its final form is introduced in 1977 (Glover, 1968a; 1968b). SS has advantages such its flexibility and sophistication which make it a visible for integration with other method to find the global optimum solution for many problems. In the field of parameter setting, a few works are done using SS and oriented to binary classification problems. For instances; Lin and Chen (2012) suggest an approach to determine the parameters and feature selection for C4.5 algorithm by employing SS meta-heuristics strategy. In another research (Chen *et al*., 2011), SS approach was used to determine the parameters of three machine learning algorithms and performing feature selection to enhance the classification accuracy. However, these works were concerned for binary classification problems. Generally, the optimization algorithms such as GA, SS, ACO and PSO are a good approach to employ for optimal search in several fields and investigate their ways for direct mining in categorization process as a rule based categorization algorithm (Afif *et al*., 2020). In addition; they can be used for guidance of data mining functions to determine the best solution for functional parameters values and relevant features for predictions. They can produce several alternative parameters setting and/or multi relevant subset of prediction features through reproduction operations on its behavior to finding the best values for parameters and/or best solution for optimal features of specific search problem (Lin *et al*., 2008a; Ghareb *et al*., 2016).

The Proposed Method

The main objective of this study is to enhance SVM performance by employing SS optimization approach for automatic setting of the SVM parameters specifically with multiclass prediction problem for Lung Cancer Diagnosis after validation with standard benchmark datasets. Thus the proposed method depends on three major components; SVM with Kernel function, decomposition strategy of multiclass and SS for parameters tuning optimization. The next subsections present the major components of proposed method.

SVM and Kernel Function

The SVM concept is discussed earlier in section 2, more details can be found in (Tharwat, 2019; Joachims, 2002; Zhang *et al*., 2015). The proposed method uses Gaussian kernel as kernel function that assist SVM to find best separation of different classes and achieves the optimal solution. The parameter of kernel function is optimized along with other SVM parameters using SS. In this study the Gaussian kernel is used because the linear kernel has been proven to be a special case of the Gaussian kernel. Also, it has few parameters rather than other kernels and it is usually numerically more stable than both polynomial and sigmoid kernels (Tharwat, 2019; Keerthi and Lin, 2003; Yin and Yin, 2016; García-Pedrajas and Fyfe, 2008). It should be noted that any other kernel function can be used and optimized but the complexity will be differ. The Gaussian kernel, $K(X, Z)$, is illustrated in Equation (5), where mapping is introduced from input space into higher dimension feature space and σ is the kernel parameter:

$$K(X,Z) = \exp\left(\frac{-\|X-Z\|^2}{2\sigma^2}\right) \quad (5)$$

Multiclass Decomposition Strategy

Solving multiclass classification problem is still an on-going research issue. There are two approaches are employed to solve multiclass classification problems using SVM. The first one includes modifying the SVM learning algorithm, often this type of modifications are not trivial and may produces costly learning algorithm (Hsu and Lin, 2002b). Second approach is the decomposition, which depends on splitting the multiclass problem into several binary subtasks. This is more frequent and common used approach. Many decomposition methods are suggested in literature. These approaches can be divided into two groups: the code matrix based approach and the hierarchical approach (Lorena and De Carvalho, 2008). This paper is only concerned with code matrix based approach. The code matrix based strategy is generally represented by matrix $\vec{M}$ with dimension $K \times L$, where k the number of classes and L represents the number of binary classifiers that is required in multiclass solution. Every row of the matrix M has a binary code associated with one of the classes. The columns of M define binary partitions of the K classes and correspond to the labels that these classes assume the binary classifiers generation. Each element of M has value in the set −1, 0, +1. If element m_{ij} equal to +1 means that the corresponding class to row i has positive label in the induction of the classifier f_j, while value -1 represent negative label and 0 for instances from class i that do not include or involve in the classification process of the classifier f_j. A new data $\vec{x}$ will be classified by applying the decoding process, which based on evaluating the predictions of the l classifiers, that generate a vector $\vec{f}(\vec{x}) = \left(f_1(\vec{x}), \cdots, f_1(\vec{x})\right)$. Then, this vector is compared to $\vec{M}$ rows. The pattern is assigned to the class whose row is closest according to some integration, researchers distance measure. Several decoding functions that can be employed in the SVM binary classifiers integration, for example; researchers in (Allwein *et al*., 2000) suggested the use of a decoding function depends on the margins by which the instance is classified by the binary SVM. This was the function that employed by (Lorena and De Carvalho, 2008) and this study also will be based on that function in decoding process, where it is equation is given below:

$$d_M\left(\vec{m}_q, f(\vec{x})\right)=\sum_{i=1}^{l}\max 0,1-\left(m_{qi}\times f_i\ (\vec{x})\right) \tag{6}$$

where, $\vec{m}_q$ represent the q^{th} row of M matrix, while $q = 1,\cdots, k$. The most popular approaches for decomposition are: One Against One (OAO), One Against All (OAA) and Error Correcting Output Codes (ECOC) (Lorena and De Carvalho, 2008; Dietterich and Bakiri, 1994). This paper focuses only on OAO and OAA approaches.

As discussed in related works (Lorena and De Carvalho, 2008; Dietterich and Bakiri, 1994), in the OAO decomposition approach, a number of binary classifiers ($k(k-1)/2$) are generated to classify datasets with k-classes where $k>2$. Every classifier is used to separate one pair of data classes (i,j), where ($i \neq j$). The code matrix in this case has dimension $k \times k(k-1)/2$. In a column representing the pair (i,j), the value of the element in i row is +1 and the value of the of the member in j row is -1, while all other elements in the column have 0 value, means that examples from the other classes do not include in this classifier. Figure 1 shows an OAO matrix for problem with four classes.

Likewise; in the OAA decomposition approach, k binary classifiers are generated to classify datasets with k-classes where $k>2$. Each classifier is trained to separate class i against reset classes. The representation of code matrix in this strategy is given by a matrix with $k \times k$. All elements in the diagonal of the matrix have value +1, while -1 value for the reset elements of the matrix. Figure 2 shows OAA matrix for problem with four classes.

Solution Representation

In this study, the solution is represented as a vector with dimension equals to the number of trial solutions. Figure 3 depicts the solution representation, where P_1 σ is kernel parameter while others are SVM parameters, P_2 C is Complexity, $P_{3\epsilon}$ is epsilon and P_4 t tolerance. The accuracy rate of every binary classifier is used to measure the quality of solution, which called the fittness function (fit). Accuracy rate for binary and multiple classes is calculated as given in Equation (7) (Huang and Wang, 2006; Afif *et al*., 2020; Dietterich and Bakiri, 1994):

$$Accuracy=\frac{TP+TN}{TP+TN+FP+FN} \tag{7}$$

where, True Positive (TP) is the positive cases that classified correctly as positive, True Negative (TN) is the negative cases that classified correctly as Negative, while, False Positive (FP) some cases with negative class classified as positive and False Negative (FN) are the cases with positive class classified as negative.

Scatter Search for Parameters Tuning

SS technique, unlike the most of other evolutionary algorithms, uses a small set of best solutions called a reference set which updated frequently during execution. The basic steps and components of SS can be described as follow (Chen *et al*., 2011; Glover, 1977; Lin and Chen, 2012):

I. *Diversification Generation Method:* Generates population *POP* using a diversification scheme
II. *Improvement Method:* Refines the new produced solutions
III. *Reference Set* (RS) *Update Method:* Chooses an initial RS from population *POP* and updates the RS in each iteration
IV. *Subset Generation Method:* Generates subsets of RS as an initial stage to produce new combined solutions
V. 5. *Solution Combination Method:* Produces one new combined solution or more from each subset generated by Subset

$$\begin{bmatrix} +1 & +1 & +1 & 0 & 0 & 0 \\ -1 & 0 & 0 & +1 & +1 & 0 \\ 0 & -1 & 0 & -1 & 0 & +1 \\ 0 & 0 & -1 & 0 & -1 & -1 \end{bmatrix}$$

Fig. 1: Code matrix of OAO method

$$\begin{bmatrix} +1 & -1 & -1 & -1 \\ -1 & +1 & -1 & -1 \\ -1 & -1 & +1 & -1 \\ -1 & -1 & -1 & +1 \end{bmatrix}$$

Fig. 2: Code matrix of OAA method

$$\begin{bmatrix} P_1 & P_2 & P_3 & P_4 \\ \cdot\cdot\cdot & \cdot\cdot & \cdots & \cdots\cdot\cdot \\ P_{1n} & P_{2n} & P_{3n} & P_{4n} \end{bmatrix}$$

Fig. 3: Solution representation

Generation Method

In this study, the diversification generation step depends on generating random values for all parameters in the solution representation. Equation (8) is used to generate solutions; solutions mean the parameters values that need to tune it in this method. The number of generated solutions called Populations Size (PoSize) and PLIST represent the pool for saving solutions:

$$X_{[sol][i]} = LB[i] + \left(UB[i] - LB[i]\right) \times Rand \qquad (8)$$

where, $X_{[sol][i]}$ is the solution number *sol* for parameter number *i*, *LB[i]* is the lower bound of the parameter number *i*, *UB[i]* is the upper bound of the parameter number *i* and *Rand* is a random value between (0,1). For example, assume that, *LB*[0] = 0.001, *UB* [0] = 32 and Rand = 0.03, X0 = 0.001 + (32-0.001) ×0.03 = 0.96097. This example illustrates how Equation 8 invokes to generate the solution. In addition, this method will be called for each binary classifier because the method depends on solutions generation for each binary classifier. After that, the solutions that generated are used for model building and testing. From these solutions the primary reference set is build, by selecting the solutions that produce good accuracy from PLIST. The number of solutions called RS size. After that, Subset generation method is invoked to generate all subsets of solutions in the RS, to be suitable for combination to generate new solutions. The following example illustrates how this steps is performed where the maximum number of subsets is: RS-Size × (RS-Size-1)/2. Assume that solutions are ×1, ×2, ×3, ×4, ×5 and RS-size = 5, then maximum number of subsets = 10 and the subsets are:

$$\begin{Bmatrix} (x_1,x_2),(x_1,x_3),(x_1,x_4),(x_1,x_5),(x_2,x_3), \\ (x_2,x_4),(x_2,x_5),(x_3,x_4),(x_3,x_5),(x_4,x_5) \end{Bmatrix}$$

In the next step, the solutions combination is performed, where a number of new solutions are generated from each subset that generated in the previous step as follows:

$$C_1 = P_1 + \left(P_2 - P_1\right) \times r_1, \qquad (9)$$

$$C_2 = P_2 + factor \times \left(P_2 - P_1\right) \times r_2, \qquad (10)$$

$$C_3 = P_1 - factor \times \left(P_2 - P_1\right) \times r_3, \qquad (11)$$

where, r_1, r_2 and r_3 are random numbers in (0,1) and factor = 0.5. Means that three solutions will be generated from each subsets that generated, if the number of subset = 6; then 18 solutions will be generated. After that, solutions are used for model building and testing and the results will be saved in pool together with solutions in the RS in order from the best one to worst. Then, the RS is updated to has the high quality solutions; RS-Size1 solutions from the pool and the RS-Size2 diverse solutions where RS-Size1+RS-Size2 = RS-Size. The diverse solution is selected based on calculating the Euclidean distance for each solution in the RS and solutions in pool. The RS-Size2 solutions with the maximum distance are selected as diverse ones. The subset generation, solution combination and RS update steps are repeated to find the best solutions in an iterative procedure until one of the termination conditions is satisfied. This research paper defines three termination conditions, therefore the termination is activated, the optimized solutions are retrieved and the process is stopped when any of the following conditions is satisfied:

i. All possible solutions for parameters value are generated for a given interval, or
ii. The achieved accuracy rate is 100% by at least one solution after validation, or
iii. The maximum number of iterations ($Max_{Iteration}$) is reached

Experimental Results and Analysis

This section presents the experiments that conducted to validate and test the performance of the proposed integration method of SVM with SS for parameter tuning. The performance is traced and measured on several datasets in term of classifier performance, error rate and standard deviation of error rate. In addition, a comparison between the proposed method and other related methods is conducted to show the efficiency of proposed method comparing to other methods in terms of classification accuracy. Furthermore, it shows the effectiveness of SS as parameter tuning algorithm and shows its effect on SVM performance as classification method for several datasets.

Datasets and Experiments Setting

The proposed method is evaluated on two types of experimental datasets from different domains, the first domain includes 9 datasets and the second experiment is conducted on lung cancer datasets. The method performance is traced and measured on these datasets and a comparison with some related works is provided. In the first experiments, as illustrated in Table 1, nine datasets from LibSVM tool webpage (Lin and Chang, 2011) are used to verify the quality of the proposed method. In addition, Tables 2 and 3 summarize all parameters setting used in the proposed method with their assigned values. These chosen values are based on the common setting in the literatures (Chen *et al*., 2011; DeCoste and Wagstaff, 2000; Williams *et al*., 2007; Lin and Lin, 2003) and based on the conducted numerical experiments.

Table 1: Datasets information and distribution

Dataset	ID	Features	Instance	Classes
DNA	DN	180	2000	3 class
Glass	GL	9	214	6 class
Iris	IR	4	150	3 class
Svmguide2	SV	20	391	3 class
Segment	SE	19	2310	7 class
Satimage	SA	32	3104	6 class
Vowel	VO	10	528	11 class
Vehicle	VE	18	846	4 class
Wine	WI	13	178	3 class

Table 2: SVM parameters value range

Parameter	Symbol	Interval
P_1	σ	[0.0001, 33]
P_2	C	[0.01, 35000]
P3	€	[0.00001, 0.0001]
P4	t	[0, 0.5]

Table 3: Parameters setting of SVM and kernel

Parameter	Def.	Val.
PLIST	Size of population	25
RS-Size	Size of initial Reference Set	5
RS-Size$_1$	Number of Best Solutions	4
RS-Size$_2$	Number of Diverse Solution	1
Maximum$_{Iteration}$	Max number of Iteration	100

Results and Discussion on 9 Datasets from UCI

To guarantee valid results for making predictions regarding new data, the proposed approach use the holdout method, which is the simplest testing technique that avoids over-fitting problem (Hamel, 2011). The holdout method depends on splitting the datasets into two parts; one for training and the other for testing with size 70% and 30%, respectively. The results are listed in Table 4 and Table 5 for OAO and OAA methods on 9 datasets. Each table contains the accuracy rate for training (Acc. Training) and remainder columns contain: Accuracy rate for testing process (Acc. Testing), the number of generation when the best solution is obtained (No.Gen.Best Sol.), number of hitting the best solution (No.Hit.Best Sol.) and fitness function evaluation times (Fitness ET). The average of accuracies that achieved in training and testing phase using OAO method are 96.41%, 97.89% and the maximum and minimum are 100%,100%, 83.24%, 91.94% respectively, while the standard deviation are 6.19% and 3.18%. Moreover, the accuracy rate for training and testing phase verify the approach does not suffer from the over-fitting and under-fitting problem. This can be noted via the differences between the accuracy rate for training and testing, where the maximum difference is 9.37% and the minimum is -1.66% with 1.46% and 3.27% for the average and standard deviation, respectively for OAO method. While in the OAA method, the maximum is 5.98% and -0.31% for minimum difference, with average 0.99% and 2.52% for standard deviation. These differences for two methods are very reasonable and according to the fact that there is no large difference between the training and testing accuracy (Chen *et al*., 2011; Lin and Chen, 2012). Figures 4 and 5 depict these differences graphically.

Furthermore, Table 6 and Table 7 list other aspects of performance measures for the proposed method. The tables list the Error Rate for Testing (ER.Rate. TS), which is calculated by dividing the sum of errors over the times of classification or over the number of required classifiers (Lorena and De Carvalho, 2008). The second column displays the Standard Deviation of error rate for Testing (StDev. TS), the reminder columns contain in sequences the sensitivity and specificity, where they reflect the true positive rate and true negative rate, expressed as a percentage, respectively. The sensitivity and specificity also reflect how well the classifier discriminates between case with positive and with negative classes (Huang and Wang, 2006; Hamel, 2011). The last two columns list Error Rate for Training phase (ER.Rate TR) and the Standard Deviation (StDev. TR), where error rate is calculated by dividing the summing of errors for all classifiers over the number of classifiers. The average of (ER.Rate. TS) and (ER.Rate TR) for OAO method are 1.55%, 0.0358, while sensitivity and specificity that produce are 98.02%, 93.74% with standard deviation 1.92% and 10.8%, respectively. From the results, we can conclude that the outcomes of the approach are encouraged for the two methods OAO and OAA, where OAO is the best in all the performance aspects. Furthermore, it is also faster in training and seems preferable for problems with a large number of classes (Galar *et al*., 2011; Milgram *et al*., 2006). Additionally, the size of datasets that are used were differ because some of datasets have a large number of instances, which require more time especially with OAA method.

Table 4: Results using OAO method

ID	Acc. training	Acc. testing	No.Gen.Best Sol.	No.Hit. Best Sol.	Fitness ET
IR	99.51	100.00	3	128	210
GL	83.24	92.61	453	362	16270
SV	88.3	91.94	300	0	10560
VO	100.00	99.93	205	2523	8605
WI	99.59	100.00	3	155	180
DN	99.74	98.08	300	0	10560
VE	98.10	98.81	204	48	7310
SE	99.93	99.85	318	860	11640
SA	099.27	099.56	1083	164	38260

Table 5: Results using OAA method

ID	Acc. training	Acc. testing	No.Gen.Best Sol.	No.Hit. Best Sol.	Fitness ET
IR	98.74	100.00	3	65	210
GL	77.86	83.84	502	2	17725
SV	88.64	90.67	300	0	10560
VO	99.85	99.54	247	162	8970
WI	100.00	100.00	5	86	280

Table 6: Results using OAO method

ID	ER. Rate.TS	StDevTS	Sensitivity	Specificity	ER. Rate.TR	StDev.TR
IR	0	0.0000	100	100.00	0.0049	0.00849
GL	1.6	2.6939	96.55	81.72	0.1676	0.3444
SV	6.33	6.806	96.95	69.44	0.117	0.04288
VO	0.01818	0.1348	99.88	100.00	0.000	0.00000
WI	0	0.0000	100	100.00	0.0041	0.00712
DN	2.33	2.0816	97.018	97.38	0.00255	0.00443
VE	1.5	2.81	94.6	97.18	0.019	0.02835
SE	0.285	0.956	99.54	99.86	0.00069	0.00231
SA	1.933	2.6313	97.61	98.09	0.00726	0.01015

Table 7: Results using OAA method

ID	ER. Rate.TS	StDev TS	Sensitivity	Specificity	ER. Rate.TR	StDev.TR
IR	0.0000	0.000	100.00	100.00	0.01260	0.0109
GL	10.5000	23.290	61.53	96.00	0.22140	0.3874
SV	11.0000	14.930	84.74	93.64	0.11350	0.0253
VO	0.7272	1.848	94.33	99.93	0.00147	0.0049
WI	0.0000	0.000	100.00	100.00	0.00000	0.0000

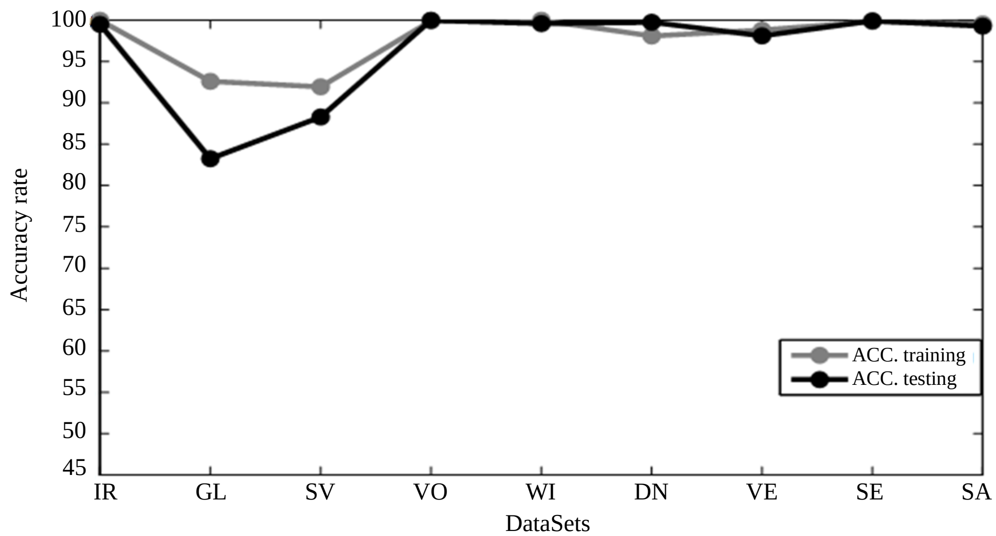

Fig. 4: Depict the accuracy of training and testing OAO

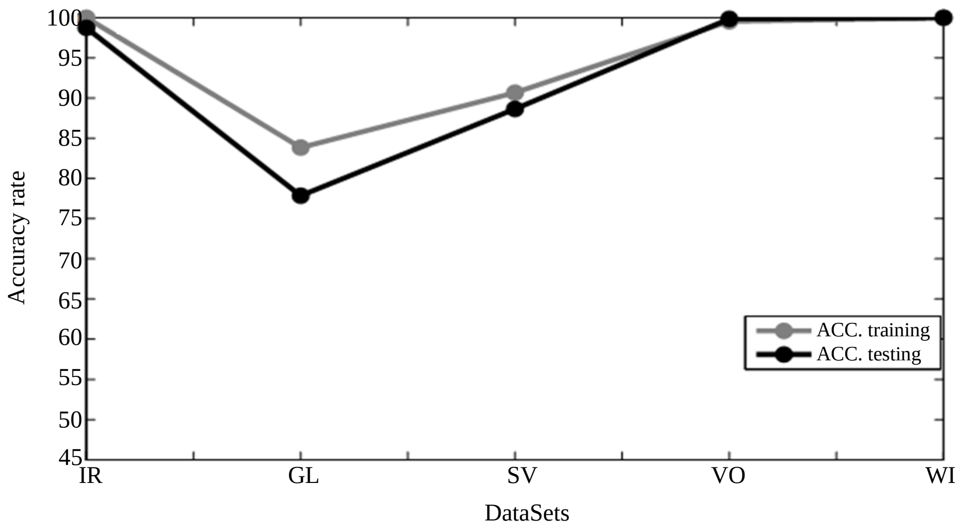

Fig. 5: Depict the accuracy of training and testing OAA

Thus, the proposed approach especially OAO performs well when using datasets with high dimensional and large number of instances. This is proved through experiments that are conducted. In order, to illustrate the performance of the proposed approach, the obtained results are compared with other published approaches (Lorena and De Carvalho, 2008; Samadzadegan *et al*., 2010; Blondin and Saad, 2010) as shown in the Tables 8 to 13. Tables 8 and 9 display comparisons between the method with others developed by (Samadzadegan *et al*., 2010), which uses GA and Grid search method to tune the parameters of SVM in multiclass decomposition. The outcomes of suggested method are the best in all methods (OAO and OAA), where the accuracy rate is increased with average are 5.95% and 10.95% in OAO method and 3.81% and 8.81% for OAA. Also, a statistical analysis is performed to prove if there any significant difference is existed in performance between the proposed method and related method (Samadzadegan *et al*., 2010). The statistical analysis for the performance of the approach is an important and necessary task to be conducted for evaluation. Statistics enable us to determine if there are any significant differences among the results produced by the suggested method when comparing with other related methods. Some of researches recommended that the use of nonparametric tests is good to show significant differences (Galar *et al*., 2011; Demšar, 2006; García *et al*., 2009; 2010; Garcia and Herrera, 2008). The Wilcoxon test is used to compare the outcomes; Table 10 reports the results of statistical analysis. The p value are 0.042, 0.043 for OAO method. This means that the significant differences is exist, where the produced *p*-value less than 0.05. Moreover, there are some differences between the proposed method and method suggested by (Samadzadegan *et al*., 2010) that uses the same values of the parameters for all binary classifiers in two method OAO and OAA. Also the method does not uses the code matrix decomposition strategy.

Tables 11 and 12 list the comparisons between testing error rate and standard deviation of the proposed method and method developed by Lorena and De Carvalho (2008), where the code matrix were used by two methods. The major difference is that their suggested method divide the datasets into two groups; one part for validation and the other for testing using hold out method and cross validation fold method. Also, the method uses GA for tuning SVM parameters for the binary classifiers in decomposition strategy. There are three groups of experiments are performed, in the first one using the same values of parameters for all binary classifiers and in the second the GA generate values for every binary classifier in decomposition and the last one uses the default values of parameters as in LibSVM tool.

Error rate for testing or for validation is the measure, which riles on it to evaluate the performance. This measure can be describe as an indicator of the ability of the method in assign each element or instance to its valid category or class. Thus, the major goal is to minimize the error rate of testing when using the code matrix strategy for classifying multiclass datasets. Tables 11 and 12 show the comparisons in cases of using different values for SVM parameters in decomposition. The suggested approach minimizes error rate (ER.Rate.TS) with average 1.98, 5.37 and the standard deviation are 2.99, 6.21 for OAO and OAA methods respectfully, comparing to method developed by (Lorena and De Carvalho, 2008), where their average of error rate are 13.85, 13.13 with standard deviation 13.22 and 12.48.

Table 8: Comparison of the proposed method (using OAO) and (Samadzadegan *et al*., 2010)

ID	Proposed Method	GA-SVM Samadzadegan *et al*. (2010)	Grid Search Samadzadegan *et al*. (2010)
IR	100.00	97	92
GL	92.92	79	70
SV	91.94	86	83
VO	99.93	97	93
WI	100.00	96	92
VE	98.81	−	−
DN	98.08	−	−
SE	99.85	−	−
SA	99.56	−	−

Table 9: Comparison of the proposed method (using OAA) and (Samadzadegan *et al*., 2010)

ID	Proposed Method using OAA	GA-SVM (Samadzadegan *et al*., 2010)	Grid Search (Samadzadegan *et al*., 2010)
IR	100	96	90
GL	83.84	78	71
SV	90.67	86	84
VO	99.54	96	93
WI	100	96	91

Table 10: Wilcoxon test Results when comparing the proposed method (using OAO) and other methods

Method	Min Acc.	Max Acc.	Avg Acc	Std.Dev.	R+	R-	P value
Proposed method		91.94	100	96.95	4.15	5	00.043
GA-SVM (Samadzadegan *et al*., 2010)		79.00	97	91.00	8.15	0	50.043
Proposed Method		91.94	100	96.95	4.15	5	00.042
Grid Search (Samadzadegan *et al*., 2010)		70.00	93	86.00	9.82	0	50.042

Table 11: Comparison between the proposed method using OAO and Lorena and De Carvalho (2008)

	Proposed Method (SS-SVM)		GA method Lorena and De Carvalho (2008)			
ID	ER.Rate.TS	StDev	ER.Rate.TS.	StDev	ER.Rate.VL.	StDev.
IR	0.00000	0.0000	4.7	4.5	1.8	2.4
GL	1.60000	2.6900	30.9	9.3	20.8	8.7
SV	6.33000	6.8060	17.6	4.4	13.6	2.4
WI	0.00000	0.0000	2.2	5.4	0.2	0.7
VO	0.01818	0.1348	−	−	−	−
DN	2.33000	2.0800	−	−	−	−
VE	1.50000	2.8100	−	−	−	−
SE	0.28570	0.9560	−	−	−	−
SA	1.93000	2.6300	−	−	−	−

Table 12: Comparison between the proposed method using OAA and Lorena and De Carvalho (2008)

	Proposed method (SS-SVM)		GA method Lorena and De Carvalho (2008)			
ID	ER.Rate.TS	StDev	ER.Rate.TS.	StDev	ER.Rate.VL.	StDev.
IR	0.0000	0.000	4.7	4.5	1.5	1.8
GL	10.5000	23.290	29.5	5.6	22.0	5.6
SV	11.0000	14.930	16.1	4.5	14.1	2.5
WI	0.0000	0.000	2.2	5.4	0.6	1.4
VO	0.7272	1.848	−	−	−	−

This means that the performance of the method is satisfactory for code matrix strategy. Table 13 shows the comparisons with method developed by (Hsu and Lin, 2002b), which utilize some meta-heuristics approaches and grid search. The method uses only multiclass datasets with the suggested method. Statistical analysis on outcomes is performed using Wilcoxon test, Table 14 illustrates the produced results. Although, there are no significant differences found, but the proposed method win in achieving the better accuracy rate for all datasets from the with increasing rate 4.55% 4.30% and 4.14% from the Grid search, PSO and APS-SVM methods found in (Blondin and Saad, 2010). Thus, proposed method gives comparable outcomes than the method developed by (Blondin and Saad, 2010) as noted through the statistical results. From comparisons and statistical analysis of results in the previous tables, one may conclude that the obtained results by the method is very encouraged relatively to some other published methods. Moreover, the experimental results prove the proposed method is an effective approach for tuning SVM parameters in code matrix decomposition strategy form other methods, this may due to using SS to search for near optimal values of SVM parameters, where it is success to explore the all possible search space to extract and maintain the best values for parameters that enhance or improve the performance of SVMs classifiers in multiclass. This enhanced the overall performance of the method as shown in the previous sections. Also, the method can deal with high dimensional and large datasets, where the number of datasets that are used are nine datasets and the maximum number of classes is 11.

Results Analysis of Lung Cancer Diagnosis

The efficiency of the proposed method is validated in the previous section, where its successfully applied and experimented on nine datasets and produced promising results. As discussed throughout this paper, one major objective of this paper is the prediction of Lung cancer using the proposed method, which enhanced and validated for this purpose. Therefore; in this section, the method is applied for diagnosis of Lung cancer disease and its performance is investigated. The experimental datasets of lung cancer are obtained from UCI machine learning repository. The datasets contained 32 instances distributed into three classes which represent three types of lung cancer. There are 57 features (attributes) for each sample, where their values are arranged form 0 to 3. This datasets are mainly used for assistance of cancer diagnosis and to predict the cancer type.

For applying, the proposed method on Lung cancer datasets, some preprocessing procedures are applied which improve representation of this datasets for mining phase; i.e., prediction of cancer type using the parameter values generation of SVM. First, there are few missing values for some attributes; all missing attributes are replaced by the mean value of the corresponding attribute for all samples of the same class. After that, data normalization is required in order to prevent feature values in greater numeric ranges from dominating and to avoid numerical difficulties during the calculation. The normalization step is performed using Equation No. (12) as given below; where the data is normalized based on the minimum ($X_{\min}$) and maximum ($X_{\max}$) values of features:

$$X_{Normalized} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{12}$$

In addition, a new setting of the proposed method parameters is applied for this case study; where a new range from 0.1 to 10000 is specified and the maximum number of iterations was set to 75. The result is presented in Table 15 in terms of accuracy rate of training and testing datasets. As shown in this table, the highest accuracy is achieved and the best solution is reached in generation No. 82. The best solution is reached four times during the evaluation times of fitness Function (FT).

The produced results compared with other available results in the same field. Table 16 shows the comparisons of more than 16 methods proposed in literature as listed in (Polat and Güneş, 2008) and (Daliri, 2012). It’s clear from comparisons that the proposed method achieves the highest accuracy rate and gives better results than the most comparable methods. It should be noted that there are some major differences with other approaches that proposed in literature; some methods perform feature reduction (Polat and Güneş, 2008) and (Daliri, 2012), where they reduced the features of the datasets into 4 and 8 features for the two methods respectively. As summarized in Table16; the achieved result of the proposed method was very promising and giving best performance relatively to related methods (Polat and Güneş, 2008; Daliri, 2012; Avci, 2012). Additionally, other related methods also produce notable results for data classification and diseases diagnosis (Afif *et al*., 2013; Afif and Hedar, 2012). Therefore, the optimized methods that based on meta-heuristic search approaches may be employed successfully to help doctors or medical specialists for diagnosis of lung cancer types and for other several diseases that should early be predicted to minimize their effect on patients.

Table 13: Testing Accuracy of the proposed method comparing with (Blondin and Saad, 2010)

ID	Proposed Method	Grid Search	PSO	APS-SVM
DN	98.08	96.13	96.49	96.42
GL	92.61	–	–	–
IR	100.00	97.47	97.87	97.93
SV	91.94	–	–	–
VE	98.81	85.90	86.11	86.67
VO	99.93	–	–	–
WI	100.00	99.16	99.21	99.33
SE	99.85	–	–	–
SA	99.56	–	–	–

Table 14: Wilcoxon test Results when comparing the proposed method and (Blondin and Saad, 2010)

Method	Min Acc.	Max Acc.	Avg Acc.	Std. Dev.	R+	R-	P value
Proposed method	98.08	100.00	99.22	0.95	4	0	0.068
Grid Search (Blondin and Saad, 2010)	85.90	99.16	94.66	5.97	0	4	0.068
Proposed method	98.08	100.00	99.22	0.95	4	0	0.068
POS by (Blondin and Saad, 2010)	86.11	99.21	94.92	5.97	0	4	0.068
Proposed method	98.08	100.00	99.22	0.95	4	0	0.068
APS by (Blondin and Saad, 2010)	86.67	99.33	94.96	5.66	0	4	0.068

Table 15: Results using OAO method

Datasets	Training accuracy %	Testing accuracy%	Gen. Best Sol.	No. Hit Best Sol.	FT
Lung cancer	100	100	82	4	6000

Table 16: Testing accuracy of the proposed method compared with other methods

Author	Method	Accuracy Rate%	Ref.
Avci	GDA-LS-SVM	96.87	Avci (2012)
Badjio and Opulent	k-NN without Reduction	37.50	
Badjio and Poulet	k-NN + Reduction	75.00	
Daliri	GA+ELM+FIS	98.85	Daliri (2012)
HenrikBostrom	DL	62.05	
HenrikBostrom	DLP	62.05	
HenrikBostrom	DL-L	64.01	
HenrikBostrom	DLP-L	64.01	
HenrikBostrom	RS	70.01	
HenrikBostrom	RSP	70.01	
Hendrickx and Bosch	k-NN	33.33	
Hendrickx and Bosch	MAXENT	39.20	
Hendrickx and Bosch	RULES	31.70	
Hendrickx and Bosch	MAXENT-H	43.30	
Hendrickx and Bosch	RULES-R-H	25.00	
Hendrickx and Bosch	RULES-A-H	34.20	
Lei Yu and Huan Liu	FCBF	87.50	
Lei Yu and Huan Liu	CORRSF	84.17	
Lei Yu and Huan Liu	RELIEFF	80.83	
Lei Yu and Huan Liu	CONSSF	84.17	
Polat and Gunes	PCA + Fuzzy Weight. Pre.+ AIRS	100.00	Polat and Güneş (2008)
Tan and Dowe,	C4.5	40.00	
Tan and Dowe,	C5	40.70	
Tan and Dowe,	Random NULL	33.30	
Tan and Dowe,	MML Oblique Tree	46.80	
The proposed method		100.00	

Conclusion

This paper employed meta-heuristic approach called SS for tuning the SVM parameters values for each binary classifier involved in multiclass decompositions. The first experiments are conducted on 9 benchmark datasets that have a high dimensional and large size. The experimental results prove that the SS is practical for finding the best setting of SVM parameters, which enhance the SVM performance. Furthermore, the method is applied for lung cancer diagnosis as a real medical classification problem. The results demonstrate that the proposed method is promising and effective method for solving this multiclass problem and it can be extended in the future for other real problems. Moreover, the results are obtained using Gaussian kernel function; the method can also be investigated with other kernel functions.

Acknowledgement

This research is self-sponsored by its corresponding authors.

Author's Contributions

Abdullah Saeed Ghareb: Contributed to analyzing all experiments, reviewing first draft of the manuscript. Organized, written and/or authored the rest drafts of the paper, prepared and analyzed some of related works, reviewed and prepared the final paper manuscript.

Mohammed H. Afif: Designed and developing the model, performed the experiments. He also prepared first draft of paper, reviewing and correction of the final paper manuscript.

Abdel-Rahman Hedar: Experimental design, contributed to results analysis.

Taysir H. Abdel Hamid: Reviewing and approved the paper.

Abdulgabbar Saif: Finalizing the manuscript.

Ethics

Authors confirm that this work is original research paper and no ethical issues behind the publication of this manuscript.

References

Afif, M., Ghareb, A., Saif, A., Bakar, A. and Bazighifan, O. (2020). Genetic algorithm rule based categorization method for textual data mining. Decision Science Letters, 9(1), 37-50.

Afif, M. H. and Hedar, A. R. (2012, March). Data classification using support vector machine integrated with scatter search method. In 2012 Japan-Egypt Conference on Electronics, Communications and Computers (pp. 168-172). IEEE.

Afif, M. H., Hedar, A. R., Hamid, T. H. A. and Mahdy, Y. B. (2013). SS-SVM (3SVM): a new classification method for hepatitis disease diagnosis. Int. J. Adv. Comput. Sci. Appl, 4.

Allwein, E. L., Schapire, R. E. and Singer, Y. (2000). Reducing multiclass to binary: A unifying approach for margin classifiers. Journal of machine learning research, 1(Dec), 113-141.

Avci, E. (2012). A new expert system for diagnosis of lung cancer: GDA—LS_SVM. Journal of medical systems, 36(3), 2005-2009.

Blondin, J. and Saad, A. (2010, August). Metaheuristic techniques for support vector machine model selection. In 2010 10th International Conference on Hybrid Intelligent Systems (pp. 197-200). IEEE.

Burges, C. J. (1998). A tutorial on support vector machines for pattern recognition. Data mining and knowledge discovery, 2(2), 121-167.

Campos, V., Glover, F., Laguna, M. and Martí, R. (2001). An experimental evaluation of a scatter search for the linear ordering problem. Journal of Global Optimization, 21(4), 397-414.

Chen, S. C., Lin, S. W. and Chou, S. Y. (2011). Enhancing the classification accuracy by scatter-search-based ensemble approach. Applied soft computing, 11(1), 1021-1028.

Cherkassky, V. and Ma, Y. (2004). Practical selection of SVM parameters and noise estimation for SVM regression. Neural networks, 17(1), 113-126.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. Machine learning, 20(3), 273-297.

Daliri, M. R. (2012). A hybrid automatic system for the diagnosis of lung cancer based on genetic algorithm and fuzzy extreme learning machines. Journal of medical systems, 36(2), 1001-1005.

DeCoste, D. and Wagstaff, K. (2000, August). Alpha seeding for support vector machines. In Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 345-349).

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. Journal of Machine learning research, 7(Jan), 1-30.

Dietterich, T. G. and Bakiri, G. (1994). Solving multiclass learning problems via error-correcting output codes. Journal of artificial intelligence research, 2, 263-286.

Faris, H., Hassonah, M. A., Ala'M, A. Z., Mirjalili, S. and Aljarah, I. (2018). A multi-verse optimizer approach for feature selection and optimizing SVM parameters based on a robust system architecture. Neural Computing and Applications, 30(8), 2355-2369.

Galar, M., Fernández, A., Barrenechea, E., Bustince, H. and Herrera, F. (2011). An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes. Pattern Recognition, 44(8), 1761-1776.

Garcia, S. and Herrera, F. (2008). An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. Journal of machine learning research, 9(Dec), 2677-2694.

García, S., Fernández, A., Luengo, J. and Herrera, F. (2009). A study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability. Soft Computing, 13(10), 959.

García, S., Fernández, A., Luengo, J. and Herrera, F. (2010). Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. Information sciences, 180(10), 2044-2064.

García-Pedrajas, N. and Fyfe, C. (2008). Evolving output codes for multiclass problems. IEEE Transactions on Evolutionary Computation, 12(1), 93-106.

Ghareb, A. S., Bakar, A. A. and Hamdan, A. R. (2016). Hybrid feature selection based on enhanced genetic algorithm for text categorization. Expert Systems with Applications, 49, 31-47.

Glover, F. (1968a). Surrogate constraints. Operations Research, 16(4), 741-749.

Glover, F. (1998b). A template for scatter search and path relinking. Lecture notes in computer science, 1363, 13-54.

Glover, F. (1977). Heuristics for integer programming using surrogate constraints. Decision sciences, 8(1), 156-166.

Hamel, L. H. (2011). Knowledge discovery with support vector machines (Vol. 3). John Wiley and Sons.

Hsu, C. W. and Lin, C. J. (2002a). A simple decomposition method for support vector machines. Machine Learning, 46(1-3), 291-314.

Hsu, C. W. and Lin, C. J. (2002b). A comparison of methods for multiclass support vector machines. IEEE transactions on Neural Networks, 13(2), 415-425.

Huang, C. L. and Wang, C. J. (2006). A GA-based feature selection and parameters optimizationfor support vector machines. Expert Systems with applications, 31(2), 231-240.

Jia, Z. Y., Ma, J. W., Wang, F. J. and Liu, W. (2011). Hybrid of simulated annealing and SVM for hydraulic valve characteristics prediction. Expert Systems with Applications, 38(7), 8030-8036.

Joachims, T. (2002). Learning to classify text using support vector machines (Vol. 668). Springer Science and Business Media.

Keerthi, S. S. and Lin, C. J. (2003). Asymptotic behaviors of support vector machines with Gaussian kernel. Neural computation, 15(7), 1667-1689.

LaValle, S. M., Branicky, M. S. and Lindemann, S. R. (2004). On the relationship between classical grid search and probabilistic roadmaps. The International Journal of Robotics Research, 23(7-8), 673-692.

Li, C., Liu, K. and Wang, H. (2011). The incremental learning algorithm with support vector machine based on hyperplane-distance. Applied Intelligence, 34(1), 19-27.

Lin, C. J. and Chang, C. C. (2011). LIBSVM: A library for support vector machines. ACM transactions on intelligent systems and technology (TIST), 2(3), 1-27.

Lin, H. T. and Lin, C. J. (2003). A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods. submitted to Neural Computation, 3(1-32), 16.

Lin, S. W. and Chen, S. C. (2012). Parameter determination and feature selection for C4. 5 algorithm using scatter search approach. Soft Computing, 16(1), 63-75.

Lin, S. W., Lee, Z. J., Chen, S. C. and Tseng, T. Y. (2008a). Parameter determination of support vector machine and feature selection using simulated annealing approach. Applied soft computing, 8(4), 1505-1512.

Lin, S. W., Ying, K. C., Chen, S. C. and Lee, Z. J. (2008b). Particle swarm optimization for parameter determination and feature selection of support vector machines. Expert systems with applications, 35(4), 1817-1824.

Lins, I. D., Moura, M. D. C., Zio, E. and Droguett, E. L. (2012). A particle swarm-optimized support vector machine for reliability prediction. Quality and Reliability Engineering International, 28(2), 141-158.

Liu, S. and Jiang, N. (2008, August). SVM parameters optimization algorithm and its application. In 2008 IEEE International Conference on Mechatronics and Automation (pp. 509-513). IEEE.

Li-Xia, L., Yi-Qi, Z. and Liu, X. Y. (2011). Tax forecasting theory and model based on SVM optimized by PSO. Expert Systems with Applications, 38(1), 116-120.

Lorena, A. C. and De Carvalho, A. C. (2008). Evolutionary Tuning of SVM Parameter Values in Multiclass Problems. Neurocomputing, 71, 3326-3334.

Maglogiannis, I., Zafiropoulos, E. and Anagnostopoulos, I. (2009). An intelligent system for automated breast cancer diagnosis and prognosis using SVM based classifiers. Applied intelligence, 30(1), 24-36.

Martí, R., Laguna, M. and Glover, F. (2006). Principles of scatter search. European Journal of operational Research, 169(2), 359-372.

Milgram, J., Cheriet, M. and Sabourin, R. (2006, October). “One against one” or “one against all”: Which one is better for handwriting recognition with SVMs?.

Pai, P. F. and Hong, W. C. (2005a). Forecasting regional electricity load based on recurrent support vector machines with genetic algorithms. Electric Power Systems Research, 74(3), 417-425.

Pai, P. F. and Hong, W. C. (2005b). Support vector machines with simulated annealing algorithms in electricity load forecasting. Energy Conversion and Management, 46(17), 2669-2688.

Pai, P. F. and Hong, W. C. (2006). Software reliability forecasting by support vector machines with simulated annealing algorithms. Journal of Systems and Software, 79(6), 747-755.

Palaniswami, M., Shilton, A., Ralph, D. and Owen, B. D. (2000). Machine learning using support vector machines.

Polat, K. and Güneş, S. (2008). Principles component analysis, fuzzy weighting pre-processing and artificial immune recognition system based diagnostic system for diagnosis of lung cancer. Expert Systems with Applications, 34(1), 214-221.

Ren, Y. and Bai, G. (2010). Determination of optimal SVM parameters by using GA/PSO. JCP, 5(8), 1160-1168.

Samadzadegan, F. and Ferdosi, E. (2012, January). Classification of Polarimetric SAR Images Based on Optimum SVMs Classifier Using Bees Algorithm. In International Conference On Intelligent Computational Systems (pp. 106-111).

Samadzadegan, F., Soleymani, A. and Abbaspour, R. A. (2010, November). Evaluation of genetic algorithms for tuning SVM parameters in multi-class problems. In 2010 11th International Symposium on Computational Intelligence and Informatics (CINTI) (pp. 323-328). IEEE.

Sartakhti, J. S., Zangooei, M. H. and Mozafari, K. (2012). Hepatitis disease diagnosis using a novel hybrid method based on support vector machine and simulated annealing (SVM-SA). Computer methods and programs in biomedicine, 108(2), 570-579.

Schölkopf, B., Tsuda, K. and Vert, J. P. (2004). Support vector machine applications in computational biology.

Sudheer, C., Shrivastava, N. A., Panigrahi, B. K. and Mathur, S. (2011, December). Groundwater level forecasting using SVM-QPSO. In International Conference on Swarm, Evolutionary and Memetic Computing (pp. 731-741). Springer, Berlin, Heidelberg.

Tharwat, A. (2019). Parameter investigation of support vector machine classifier with kernel functions. Knowledge and Information Systems, 61(3), 1269-1302.

Tuba, E. and Stanimirovic, Z. (2017, June). Elephant herding optimization algorithm for support vector machine parameters tuning. In 2017 9th International Conference on Electronics, Computers and Artificial Intelligence (ECAI) (pp. 1-4). IEEE.

Vapnik, V. and Vapnik, V. (1998). Statistical learning theory. 1st Ed Wiley. Boston, USA.

Williams, P., Li, S., Feng, J. and Wu, S. (2007). A geometrical method to improve performance of the support vector machine. IEEE Transactions on Neural Networks, 18(3), 942-947.

Wu, C. H., Tzeng, G. H., Goo, Y. J. and Fang, W. C. (2007). A real-valued genetic algorithm to optimize the parameters of support vector machine for predicting bankruptcy. Expert systems with applications, 32(2), 397-408.

Yin, S. and Yin, J. (2016). Tuning kernel parameters for SVM based on expected square distance ratio. Information Sciences, 370, 92-102.

Zhang, X., Qiu, D. and Chen, F. (2015). Support vector machine with parameter optimization by a novel hybrid method and its application to fault diagnosis. Neurocomputing, 149, 641-651.