Computers in Biology and Medicine 43 (2013) 1978–1986

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/cbm

A note on hyper ellipse method for classifying biological and medical data

Yao-Huei Huang *

Institute of Information Management, National Chiao Tung University, Hsinchu 300, Taiwan, ROC

ARTICLE INFO

Article history:
Received 7 October 2011
Accepted 15 August 2013

Keywords:
Hyper ellipse
Nonlinear program
Linearization

ABSTRACT

The classification of biological and medical datasets is essential to humanity. This study proposes a hyper ellipse method based on mixed integer nonlinear program for classifying datasets. A linearization technique with a number of piecewise line segments is used to treat nonlinear constraints, which aims to obtain an approximate optimal solution. Numerical examples are presented to demonstrate the efficacy of the proposed method.

1. Introduction

Rapid developments in computer science have highlighted the increasingly important role of data classification in daily life. This problem has elicited considerable attention in biological and medical research. One of the major issues is classification technique [4,9,7,15,19,20,22], which ascertains the specific criteria of the dataset. Each object in a dataset that contains several objects with certain attributes belongs to a specific class. The technique is used to find a classification rule that accurately describes the features of a specific class. The methods include decision tree [5,18] and hyper-plane support vector methods [6,17,20,26,32,30,34]. In order to evaluate the performance of the classifiers, Li and Chen [20] proposed three criteria as (i) accuracy: a rule that fits one class should not cover the objects of other classes. The higher the accuracy of a rule, the better the performance will be. (ii) Support: a rule that is a good fit for a class should support most of the objects in that class. (iii) Compact: a good rule should be expressed in a compact way. The smaller the number of rules used, the better the rules will be.

Two well-known methods are used to induce classification rules. First is decision tree method, which has been developed in the last decades [5,24,25]. The method is widely applied to real-world cases such as fault isolation in an induction motor [23], classifying normal and tumor tissues [33], skeletal maturity assessment [1], and proteomic mass spectra classification [12].

Taking a classification problem with two attributes is shown in Fig. 1, as an example, where "○" represents a first-class object, and "•" represents a second-class object. Fig. 1 depicts a situation where a non-linear relationship exists between objects of two classes. Fig. 2 shows that the decision tree method requires nine branch lines to classify these objects exactly.

However, decision tree method is a heuristic approach that only induces feasible rules. This method splits the data into hyper-rectangular regions using a single variable (i.e., attribute). A huge number of variables may be used to split a rule, which generate various rules to classify all objects [20].

The second one is support vector hyper plane technique [28]. The technique separates observations on different classes via different kernel functions such as linear and nonlinear. The applications are widely discussed such as selecting features and extracting rules from gene expression data of cancer tissue [8], lung cancer detection [13], and other applications [17,21,26]. In Fig. 1, for example, a support vector hyper plane method requires a critical kernel nonlinear function to discriminate the objects as shown in Fig. 3. That is, it requires computational cost to construct a kernel matrix in training step [31]. Although support vector hyper plane method with non-linear kernel function can use one support vector line, high computational cost in constructing kernel matrix to separate objects into a distinct group [31].

Alternatively, a hyper sphere method [14] can classify objects with better accuracy, support and compactness than decision tree method. However, taking the objects in Fig. 1 as an example, it is difficult to classify these objects (See Fig. 4) since we need four rules (i.e., hyper spheres). This study therefore proposes a novel hyper ellipse method with piecewise linearization technique [2] to use much fewer rules. Then the hyper ellipse model is reformulated by using piecewise linearization approach with a number of binary variables and constraints for the piecewise line segments. The error in linear approximation decreases as the number of break points used in the linearization process increases. An approximate global optimal solution can be obtained by using

* Tel.: +886 3 5712121x 57416.
E-mail address: yaohuei.huang@gmail.com

0010-4825/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.compbiomed.2013.08.007

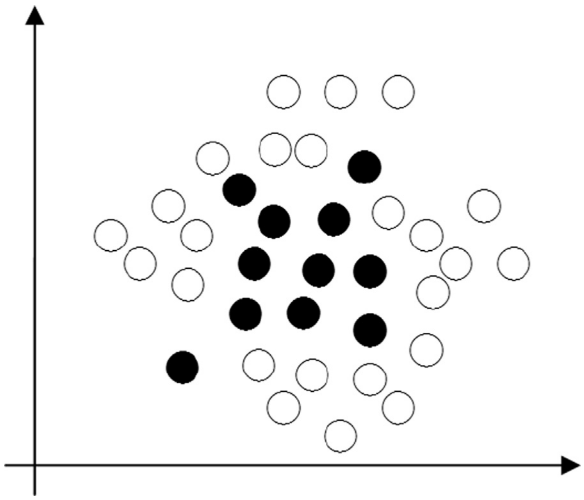

Fig. 1. A classification problem with two attributes.

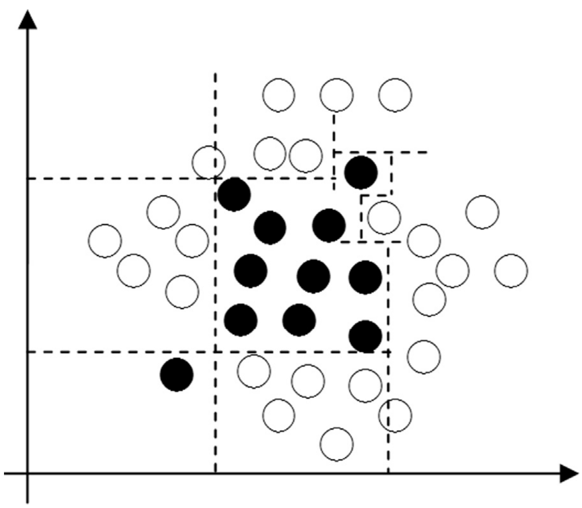

Fig. 2. Classify objects by decision tree method.

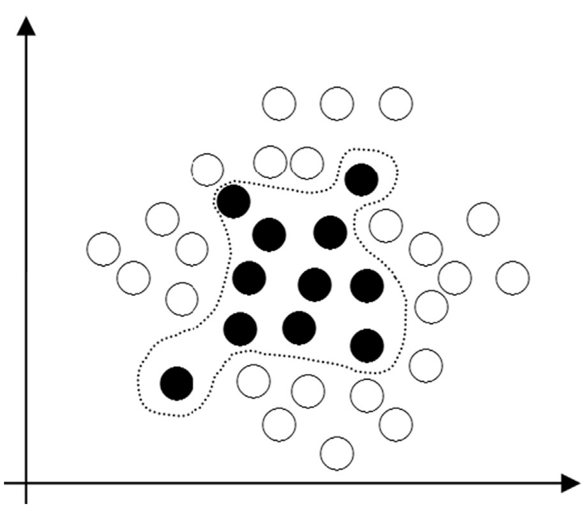

Fig. 3. Classify objects by support vector hyper plane method.

the proposed method. The concept of the hyper ellipse method is depicted in Fig. 5, where only one union of two ellipses is required to classify the objects. All objects of class "•" are covered by the union of two ellipses, and those that are not covered by this union

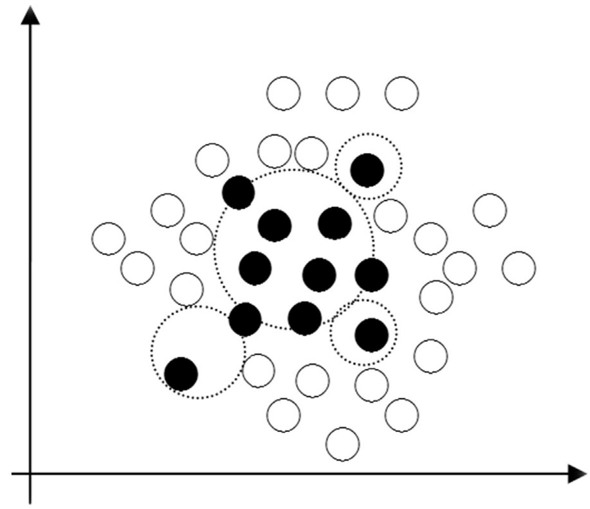

Fig. 4. Classify objects by hyper sphere method.

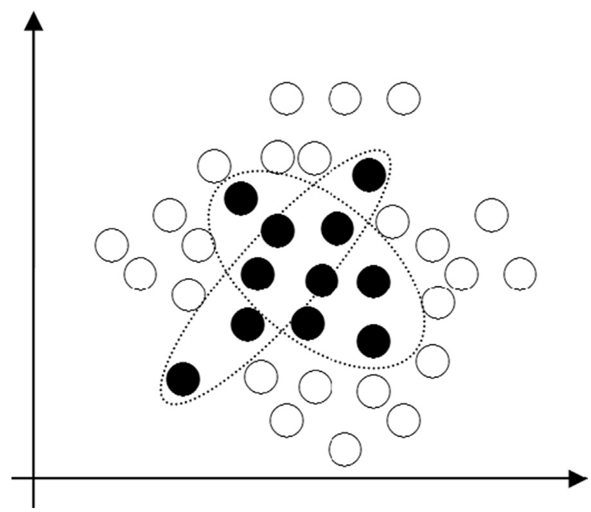

Fig. 5. Classify objects by hyper ellipse method.

belong to class "○". As a result, it derives rules with higher rates of accuracy, support, and compactness than those obtained by existing methods.

The remainder of the paper is organized as follows: Section 2 formulates a classification model of hyper ellipse; Section 3 proposes a classification algorithm; Section 4 presents numerical examples to demonstrate the efficacy of the proposed method; and Section 5 contains concluding remarks.

2. A hyper ellipse method

The three criteria for evaluating the quality of classification technique proposed by Li and Chen [20] indicate that the classification rules directly affect the rates of accuracy, support, and compactness. This study then formulates a model to find the highest rates. The following notations are used to facilitate the discussion:

$a_{i,j}$ the jth attribute value of the ith object.

$a'_{i,j}$ normalize $a_{i,j}$ as $a'_{i,j}=(a_{i,j}-\underline{a}_j)/(\overline{a}_j-\underline{a}_j)$ where $0 \le a'_{i,j} \le 1$, $\overline{a}_j$ is the largest value of attribute j; and $\underline{a}_j$ is the smallest value of attribute j.

$h_{t,k,j}$ the jth center value of the kth hyper ellipse for class t.

$b_{t,k,j}$ the jth semi-axis length of the kth hyper ellipse for class t.
$u_{t,i,k}$ decision 0–1 variables. $u_{t,i,k}=1$ if object i is covered by $E_{t,k}$; otherwise, $u_{t,i,k}=0$.
$n(t)$ the number of objects in class t.
c_i the ith object belonging to class $c_i \in \{1,\ldots,g\}$.
m the number of attributes.
R_t a rule to describe class t.
K_t a set of hyper ellipses belonging to class t, where $K_t=\{1,\ldots,k_t\}$.
$E_{t,k}$ the kth hyper ellipse for classifying objects of class t.
UE_t the number of independent hyper ellipse and unions of hyper ellipses.
$x_i=(a_{i,1},a_{i,2},\ldots,a_{i,m};c_i)$ a general form of expressing an object x_i.
$E_{t,k}=(h_{t,k,1},h_{t,k,2},\ldots,h_{t,k,m};b_{t,k,1},b_{t,k,2},\ldots,b_{t,k,m})$ a general form of expressing a hyper ellipse $E_{t,k}$.

Two and three dimensions (i.e., two attributes and three attributes) are taken as examples to illustrate an ellipse and an ellipsoid, respectively, as shown in Fig. 6. Fig. 6(a) shows the centroid of the ellipse as $(h_{t,k,1},h_{t,k,2})$, and the semi-axes of the ellipse as $(b_{t,k,1},b_{t,k,2})$. Fig. 6(b) depicts the centroid of the ellipsoid as $(h_{t,k,1},h_{t,k,2},h_{t,k,3})$, and the semi-axes of the ellipsoid as $(b_{t,k,1},b_{t,k,2},b_{t,k,3})$. Increasing the number of dimensions to m (i.e., m attributes), where $m>3$, yields a hyper ellipse.

This study considers the following non-linear model as a classification program, where the accuracy rate is fixed to 1.

Hyper ellipse model

Maximize

$$\sum_{i\in I^+} u_{t,i,k} \tag{1}$$

subject to

$$\sum_{j=1}^{m}[(a'_{i,j}-h_{t,k,j})^2/b_{t,k,j}^2]\le 1+M(1-u_{t,i,k}) \quad \forall i\in I^+, \tag{2}$$

$$\sum_{j=1}^{m}[(a'_{i',j}-h_{t,k,j})^2/b_{t,k,j}^2]>1 \quad \forall i'\in I^-, \tag{3}$$

where $u_{t,i,k}\in\{0,1\}\ \forall i\in I^+$, M is a large enough constant, $h_{t,k,j},b_{t,k,j}\ge 0$, and the sets, I^+ and I^- are defined as

$$I^+=\{i|i=1,2,\ldots,n, \text{ where object } i\in \text{ class } t\}, \tag{4}$$

$$I^-=\{i'|i'=1,2,\ldots,n, \text{ where object } i'\notin \text{ class } t\}. \tag{5}$$

Objective (1) states that objects for all $i\in I^+$ could be covered by hyper ellipse $E_{t,k}$ as much as possible. In constraint (2), if $u_{t,i,k}=1$, then object i is covered by hyper ellipse $E_{t,k}$, where $i\in I^+$. Constraint (3) enforces that object cannot be covered by hyper ellipse $E_{t,k}$, where $i\in I^-$.

Li and Chen [20] indicate that the rates of accuracy, compactness, and support for R_t in the hyper ellipse model are defined as follows:

Definition 1. The accuracy rate of a rule R_t in the hyper ellipse model is fixed as $AR(R_t)=1$. That is, the rule that fits one class never covers the objects of other classes.

Definition 2. The support rate of rule R_t in the hyper ellipse model is specified by the following items:

(i) If $\sum_{k\in K}u_{t,i,k}\ge 1$ belongs to class t, then $U_{t,i}=1$; otherwise $U_{t,i}=0$ for all i and t, where K indicates the hyper-ellipses set in class t.
(ii) $SR(R_t)=\sum_{i\in I^+}U_{t,i}/n(t)$,

where $n(t)$ indicates the number of objects belonging to class t.

Definition 3. The compact rate of a set of rules $(R_1,\ldots,R_g)$ is defined as

$$CR(R_1,\ldots,R_g)=g/\sum_{t=1}^{g}UE_t. \tag{6}$$

where UE_t represents the number of hyper ellipses and unions of hyper ellipses for class t

A union of hyper ellipses indicates that the object is covered by different hyper ellipses, as shown in Fig. 7. Furthermore, the objects of class "○" are covered by two ellipses (i.e., $E_{1,1}$ and $E_{1,2}$) and a union of three ellipses (i.e., $E_{1,3}\cup E_{1,4}\cup E_{1,5}$), and the objects of class "•" are covered by one union of two ellipses (i.e., $E_{2,1}\cup E_{2,2}$) such that $UE_1=3$, $UE_2=1$, and $CR(R_1,R_2)=2/4$.

Finding a global solution for a hyper ellipse model that maximizes a linear objective function subject to a set of nonlinear non-convex constraints is difficult. This study utilizes a piecewise linearization technique to convert a hyper ellipse model into a mixed-integer linear program, which can be solved by a global solution. The following proposition is considered:

Proposition 1. *$L(f_1(x_1,\ x_2))$ is denoted as an approximate piecewise linear function of $f_1(x_1,\ x_2)$. To simplify the presentation, we assume that x_1 and x_2 have the same number of break points (i.e., $m+1$). We consider $c_{1,1}\le x_1\le c_{1,m+1}$ and $c_{2,1}\le x_2\le c_{2,m+1}$. Then, the interval*

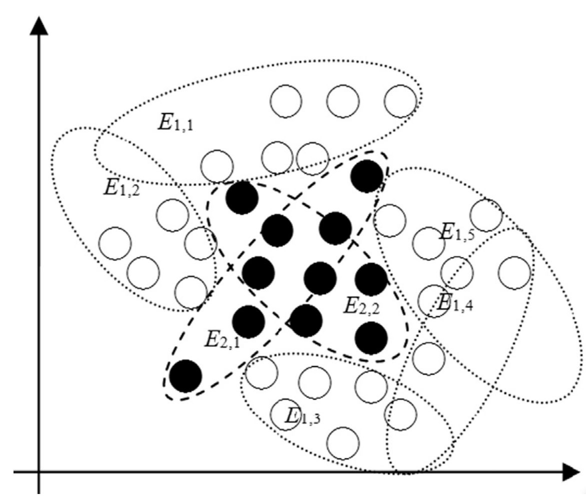

Fig. 7. Classifying objects by hyper ellipse method.

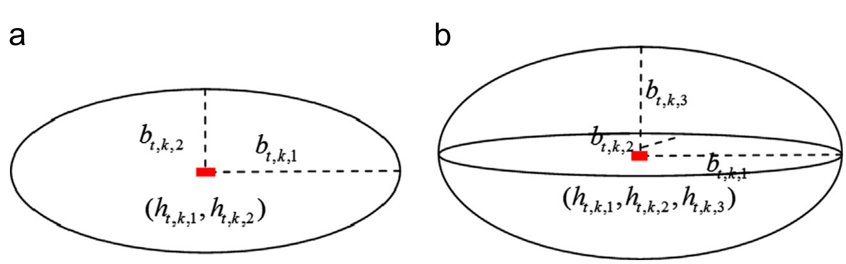

Fig. 6. The concept of hyper ellipses. (a) An ellipse, (b) an ellipsoid.

$[c_{1,1},\ c_{1,m+1}]$ is partitioned into sub-intervals via the break points (i.e., $c_{1,1},\ c_{1,2},\dots,c_{1,m+1}$), where $c_{1,1}<c_{1,2}<\dots\ <c_{1,m}<c_{1,m+1}$. Similarly, $[c_{2,1},\ c_{2,m+1}]$ is partitioned into sub-intervals with $c_{2,1}<c_{2,2}<\dots<c_{2,m}<c_{2,m+1}$. Then $L(f_2(x_1,\ x_2))$ is expressed as follows:

$$f_2(x_1,x_2)\cong L(f_2(x_1,x_2))=\sum_{k_1=1}^{m+1}\sum_{k_2=1}^{m+1} f_2(c_{1,k_1},c_{2,k_2})\omega_{k_1,k_2}, \quad (7)$$

$$x_1=\sum_{k_1=1}^{m+1}\sum_{k_2=1}^{m+1} c_{1,k_1}\omega_{k_1,k_2}, \quad (8)$$

$$x_2=\sum_{k_1=1}^{m+1}\sum_{k_2=1}^{m+1} c_{2,k_1}\omega_{k_1,k_2}, \quad (9)$$

$$\sum_{k_1=1}^{m+1}\sum_{k_2=1}^{m+1} \omega_{k_1,k_2}=1, \quad (10)$$

$$\omega_{k_1,k_2}+\omega_{k_1,k_2+1}+\omega_{k_1+1,k_2+1}\ge u_{k_1,k_2} \quad \forall k_1=1,\dots,m,\ \ k_2=1,\dots,m, \quad (11)$$

$$\omega_{k_1,k_2}+\omega_{k_1+1,k_2}+\omega_{k_1+1,k_2+1}\ge v_{k_1,k_2} \quad \forall k_1=1,\dots,m,\ \ k_2=1,\dots,m, \quad (12)$$

$$\sum_{k_1=1}^{m}\sum_{k_2=1}^{m} (u_{k_1,k_2}+v_{k_1,k_2})=1, \quad (13)$$

where $\omega_{k_1,k_2}\ge 0$ and $u_{k_1,k_2},v_{k_1,k_2}\in\{0,1\}$.

Proof (*Refer to Beale and Forrest [3]*). Constraints (10) to (11) imply that if $u_{k_1,k_2}=1$, then $\omega_{k_1,k_2}+\omega_{k_1,k_2+1}+\omega_{k_1+1,k_2+1}=1$ and $L(f_1(x_1,x_2))=f(c_{1,k_1},c_{2,k_2})\ \omega_{k_1,k_2}+f(c_{1,k_1},c_{2,k_2+1})\omega_{k_1,k_2+1}\ \ +f(c_{1,k_1+1},c_{2,k_2+1})\omega_{k_1+1,k_2+1}$. Function $f_1(x_1,\ x_2)=h_{t,k,j}^2/b_{t,k,j}^2$ is assumed to be approximately linearized as $f_1(x_1,\ x_2)\cong L(h_{t,k,j}^2/b_{t,k,j}^2)$. □

Remark 1. Both functions $f_2(x_1,\ x_2)=h_{t,k,j}/b_{t,k,j}^2$ and $f_3(x_2)=1/b_{t,k,j}^2$ can be linearized as

$$f_2(x_1,x_2)=h_{t,k,j}/b_{t,k,j}^2\cong L(h_{t,k,j}/b_{t,k,j}^2)=\sum_{k_1=1}^{m+1}\sum_{k_2=1}^{m+1} c_{1,k_1}/c_{2,k_2}^2\omega_{k_1,k_2}, \quad (14)$$

$$f_3(x_2)=1/b_{t,k,j}^2\cong L(1/b_{t,k,j}^2)=\sum_{k_1=1}^{m+1}\sum_{k_2=1}^{m+1} 1/c_{2,k_2}^2\omega_{k_1,k_2}. \quad (15)$$

This study applies the piecewise linearization technique to approximate the optimal solution in each iterative computing. The technique is to refine the hyper-ellipse region in the linearized program. This means that the more piecewise line segments are used in the hyper ellipse model, the more exact regions of hyper ellipse can be defined. This technique aims to find the high rates of accuracy, support, and compactness. Section 3 discusses an iterative algorithm that identifies high rates.

3. The proposed algorithm

The proposed classification algorithm attempts to find the highest rates of accuracy, support, and compactness by using the following procedure:

Algorithm.

Step 1. All attributes are normalized (i.e., rescale $a'_{i,j}=(a_{i,j}-\underline{a}_j)/(\overline{a}_j-\underline{a}_j)$ to be $0\le a'_{i,j}\le 1$).

Step 2. $t=1$ and $k=1$ are initialized.

Step 3. The linearized hyper ellipse model is solved to obtain the kth hyper ellipse of class t(i.e. $E_{t,k}$). Temporarily remove the objects covered by $E_{t,k}$ from class t.

Step 4. We assume that $k=k+1$ and repeat Step 2 until all objects belonging to class t have been assigned to the hyper ellipses of the same class. (Exception: Step 4 is conducted if the hyper ellipse $E_{t,k}$ only covers one object.). Finally, indicating the number of hyper ellipse $k_t=k$ in class t.

Step 5. The objects that were temporarily removed from class t in Step 2 are restored. We assume that $k=1$ and $t=t+1$, and Step 2 is repeated until all classes have been processed (i.e., for $t=1,\dots,g$).

Step 6. Object i belonging to class t is calculated whether at least one object is covered by one of the $E_{t,k}$ for $k=1,\dots,k_t$; $U_{t,i}=1$ if yes, otherwise $U_{t,i}=0$ for $t=1,\dots,g$ and $i=1,\dots,n$.

Step 7. The number of independent hyper ellipses and unions of hyper ellipses in UE_t is calculated for $t=1,\dots,g$.

Step 8. $SR(R_t)$ for $t=1,\dots,g$ and $CR(R_1,\dots,R_g)$ is output.

The optimal classification scheme to classify objects most efficiently can be designed based on Steps 1 to 7. Fig. 8 shows the flowchart of the algorithm.

4. Numerical examples

This section conducts computational experiments to evaluate the performance of the proposed model in terms of accuracy, support, and compactness rates. All test problems are solved by CPLEX [16] using the CPLEX MIP Solver to solve the corresponding mixed integer formulations of each linearized hyper ellipse model. The CPLEX MIP Solver can solve the linearized model by branch and bound solver to obtain a global optimum. All experiments are run on a PC equipped with an Intel Core i5-2430 M CPU, 8 GB RAM, and Windows 7 (64-bit) operating system.

This study uses the following three testing datasets in the experiments:

(i) The Iris flower dataset compiled by Sir Ronald Aylmer Fisher [11].

(ii) The European barn swallow (*Hirundo rustica*) dataset, which contains information obtained by trapping individual swallows in Stirlingshire, Scotland between May and July 1997 [4,20].

(iii) The HSV (highly selective vagotomy) patients dataset of F. Raszeja Memorial Hospital in Poland [10,20,29].

However, according to three criteria proposed by Li and Chen [20], the data are mainly training sets where no testing sets are separated. It aims to evaluate the methods whether it is useful for the dataset. Therefore, this study compares the proposed method with decision tree method [20] and hyper sphere method [14]. Experimental reports are discussed in Sections 4.1–4.3, and a limitation of the proposed method is described in Section 4.4.

4.1. Iris flower dataset

The Iris flower dataset contains 150 objects, each of which is described by four attributes (1: sepal length; 2: sepal width; 3: petal length; 4: petal width) and classified into one of three classes (i.e., 1: setosa; 2: versicolor; 3: virginica). All objects were taken as testing dataset to solve the proposed linear program with 32 piecewise line segments. We then induce six hyper ellipses (i.e., $E_{1,1}\in$ class 1, $E_{2,1},\ E_{2,2}\in$ class 2, and $E_{3,1},\ E_{3,2},\ E_{3,2}\in$ class 3). The induced classification rules are reported in Table 1, which lists an independent hyper ellipse (i.e., $E_{1,1}$), 2 unions (i.e., $E_{2,1}\cup E_{2,2}$ and $E_{3,1}\cup E_{3,2}\cup E_{3,3}$) of hyper ellipses, centroid points and semi-axes lengths of the hyper ellipses. The rules are discussed as follows:

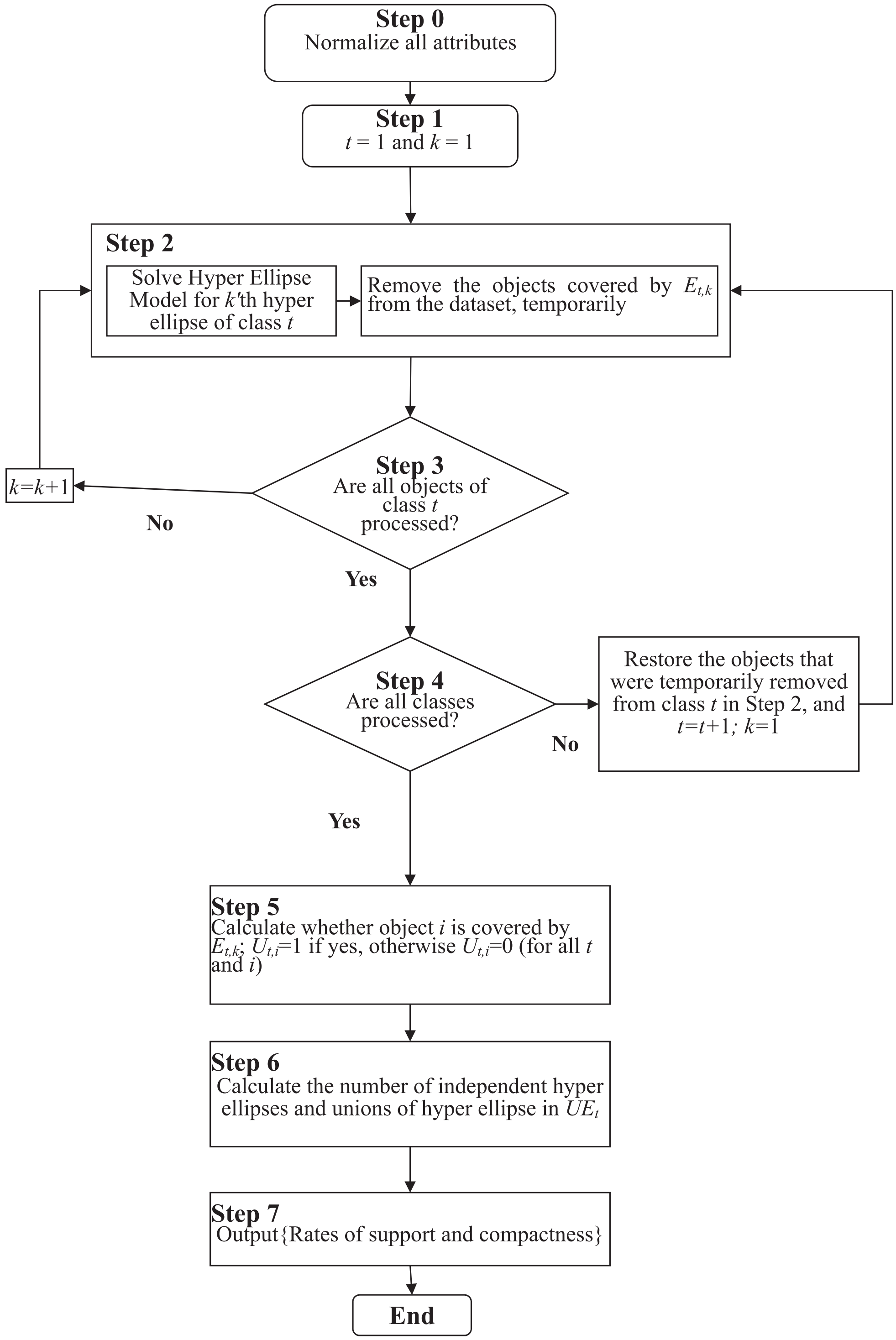

Fig. 8. Flowchart of the proposed algorithm.

Table 1
Centroid points and semi-axis lengths of the Iris dataset derived by the proposed method.

Rule #	Unions of ellipses	$E_{t,k}$	$h_{t,k,1}/b_{t,k,1}$	$h_{t,k,2}/b_{t,k,2}$	$h_{t,k,3}/b_{t,k,3}$	$h_{t,k,4}/b_{t,k,4}$
R_1	$E_{1,1}$	$E_{1,1}$	0.000/1.491	2.120/4.047	0.000/1.200	0.000/2.534
R_2	$E_{2,1} \cup E_{2,2}$	$E_{2,1}$	1.095/2.205	1.372/5.541	0.389/0.123	0.549/0.111
		$E_{2,2}$	0.306/2.425	1.608/1.798	0.666/0.109	0.637/0.017
R_3	$E_{3,1} \cup E_{3,2} \cup E_{3,3}$	$E_{3,1}$	0.000/14.398	0.000/0.879	1.219/0.717	0.960/0.185
		$E_{3,2}$	3.811/115.776	0.000/489.276	0.762/0.019	0.388/0.073
		$E_{3,3}$	0.556/2.494	0.215/0.376	0.023/1.170	1.374/0.806

Rule R_1 in Table 1 contains a hyper ellipse $E_{1,1}$, which implies that

- if $((a'_{i,1}-0)^2/1.491)+((a'_{i,2}-2.12)^2/4.047)+((a'_{i,3}-0)^2/1.2)+((a'_{i,4}-0)^2/2.534)\le 1$, then object x_i belongs to class 1. Otherwise object x_i does not belong to class 1.

Rule R_2 in Table 1 contains two hyper ellipses ($E_{2,1}$ and $E_{2,2}$), which implies that

- if $((a'_{i,1}-1.095)^2/2.205)+((a'_{i,2}-1.372)^2/5.541)+((a'_{i,3}-0.389)^2/0.123)+((a'_{i,4}-0.549)^2/0.111)\le 1$, then object x_i belongs to class 2, or
- if $((a'_{i,1}-0.306)^2/2.425)+((a'_{i,2}-1.608)^2/1.798)+((a'_{i,3}-0.666)^2/0.109)+((a'_{i,4}-0.637)^2/0.017)\le 1$, then object x_i belongs to class 2.
- Otherwise object x_i does not belong to class 2.

Rule R_3 in Table 1 contains three hyper ellipses ($E_{3,1}$, $E_{3,2}$ and $E_{3,3}$), which implies that

- if $((a'_{i,1}-0)^2/14.398)+((a'_{i,2}-0)^2/0.879)+((a'_{i,3}-1.219)^2/0.717)+((a'_{i,4}-0.96)^2/0.185)\le 1$, then object x_i belongs to class 3, or
- if $((a'_{i,1}-3.811)^2/115.776)+((a'_{i,2}-0)^2/489.276)+((a'_{i,3}-0.762)^2/0.019)+((a'_{i,4}-0.388)^2/0.073)\le 1$ then object x_i belongs to class 3, or
- if $((a'_{i,1}-0.556)^2/2.494)+((a'_{i,2}-0.215)^2/0.376)+((a'_{i,3}-0.023)^2/1.17)+((a'_{i,4}-1.374)^2/0.806)\le 1$, then object x_i belongs to class 3.
- Otherwise object x_i does not belong to class 3.

The performance of the proposed method in deducing the classification rules for the Iris dataset is compared with that of the decision tree and hyper sphere methods. The experimental results are listed in Table 2.

The accuracy rates of (R_1, R_2, R_3) under the proposed method are (1, 1, 1), indicating that none of the objects in class 2 or class 3 are covered by $E_{1,1}$; none of the objects in classes 1 or 3 are covered by $E_{2,1}$ and $E_{2,2}$; and none of the objects in classes 1 or 2 are covered by $E_{3,1}$, $E_{3,2}$, and $E_{3,3}$. The support rates of (R_1, R_2, R_3) are (1, 1, 1), indicating that all objects in classes 1, 2, and 3 are completely supported by $E_{1,1}$, $E_{2,1}$, $E_{2,2}$, $E_{3,1}$, $E_{3,2}$, and $E_{3,3}$. The compactness rate of rules R_1, R_2, and R_3 are computed as $CR(R_1,\ R_2,\ R_3)=\frac{3}{3}=1$.

Table 2
Results of the compared methods for the Iris dataset (R_1,R_2,R_3).

Items	Proposed method	Decision tree	Hype sphere method
AR(R_1,R_2,R_3)	(1, 1, 1)	(1, 1, 1)	(1, 0.98, 0.96)
SR(R_1,R_2,R_3)	(1, 1, 1)	(1, 0.98, 0.98)	(1, 0.96, 0.98)
CR	1	0.5	1

Table 3
Centroid points derived by the proposed method for the Swallow dataset.

Rule #	Unions of ellipses	$E_{t,k}$	$h_{t,k,1}/b_{t,k,1}$	$h_{t,k,2}/b_{t,k,2}$	$h_{t,k,3}/b_{t,k,3}$	$h_{t,k,4}/b_{t,k,4}$	$h_{t,k,5}/b_{t,k,5}$	$h_{t,k,6}/b_{t,k,6}$	$h_{t,k,7}/b_{t,k,7}$	$h_{t,k,8}/b_{t,k,8}$
R_1	$E_{1,1}\cup E_{1,2}$	$E_{1,1}$	4.733/92.574	0.328/0.727	3.785/68.012	0.169/1.001	0/4.109	0/151336.6	3.201/22.91	1.232/3.772
		$E_{1,2}$	0.654/2.645	0.766/1.443	0/1.407	0.511/1.459	0.113/0.444	1.094/1.265	0.761/2.321	0.181/0.249
R_2	$E_{2,1}\cup E_{2,2}\cup E_{2,3}$	$E_{2,1}$	0.395/8.378	1.461/1.872	0/7.590	0.996/1.752	1.484/14.486	1.489/11.904	0.220/0.844	0/3.741
		$E_{2,2}$	0.447/0.396	0.150/0.169	0.279/0.959	0.590/0.306	1.170/2.272	1.210/14.928	0.322/1.133	0/5.354
		$E_{2,3}$	0/12.315	2.663/7.296	0/15.882	1.748/12.346	0/168858.9	1.877/11.127	0.229/0.940	0/19.106

Table 4
Results of the compared methods for the Iris dataset (R_1,R_2).

Items	Proposed method	Decision tree	Hyper sphere method
AR(R_1,R_2)	(1, 1)	(1, 1)	(0.97, 1)
SR(R_1,R_2)	(1, 1)	(1, 0.97)	(0.97, 1)
CR	1	0.3	0.1

Table 5
Centroid points derived by the proposed method for the HSV dataset.

Rule #	Unions of ellipses	$E_{t,k}$	$h_{t,k,1}$	$h_{t,k,2}$	$h_{t,k,3}$	$h_{t,k,4}$	$h_{t,k,5}$	$h_{t,k,6}$	$h_{t,k,7}$	$h_{t,k,8}$	$h_{t,k,9}$	$h_{t,k,10}$	$h_{t,k,11}$
R_1	$E_{1,1}\cup\ldots\cup E_{1,7}$	$E_{1,1}$	0	0.744	0.497	0.201	1.297	0.111	0.084	0.431	0.703	−0.082	−0.435
		$E_{1,2}$	1	0.314	0.224	0.76	0.179	2	−0.165	0.039	1.604	1.134	1.265
		$E_{1,3}$	0.449	7.146	−5.112	−5.157	0.31	−6.208	−7.488	7.96	−2.253	−1.584	0.582
		$E_{1,4}$	1.635	−3.159	3.436	0.442	0.279	−3.163	−4.939	0.399	−2.296	−3.627	1.462
		$E_{1,5}$	0.191	7.644	0.326	0.064	−1.53	0.464	0.473	2.642	0.179	−0.681	0.892
		$E_{1,6}$	−0.001	−2.173	0.207	0.433	−0.155	−8.065	1.202	3.308	−0.112	0.062	1.135
		$E_{1,7}$	0.017	−0.526	0.514	0.058	−0.018	−0.079	0.44	0.12	0.835	0.025	0.793
R_2	$E_{2,1}\cup E_{2,2}\cup E_{2,3}$	$E_{2,1}$	0	−1.118	0.395	2.896	−4.037	0.254	0.178	−21.503	−0.79	0.074	0.639
		$E_{2,2}$	0.628	0.388	3.507	0.493	−0.195	−0.344	0.211	0.715	1.597	0.204	−0.716
		$E_{2,3}$	0.475	0.214	−0.111	−0.171	0.911	0.441	0.349	0.371	−0.237	−0.466	0.301
R_3	$E_{3,1}$	$E_{3,1}$	0	1.453	−0.141	3.593	−0.943	0.438	1.057	1.24	0.55	0.671	0.012
	$E_{3,2}$	$E_{3,2}$	0.18	0.406	0.577	0.609	−0.144	−0.103	0.168	−0.194	0.897	0.324	−0.123
R_4	$E_{4,1}$	$E_{4,1}$	1	0.479	−0.122	24.682	−21.093	30.932	−13.654	35.391	−24.189	0.139	39.425
	$E_{4,2}$	$E_{4,2}$	1	12.773	0.421	0.342	0.126	−0.114	0.579	1.267	0.283	0.152	−3.019
	$E_{4,3}$	$E_{4,3}$	1.021	0.351	−10.627	−5.166	0.202	1.949	0.063	−5.886	0.089	−6.09	6.597
	$E_{4,4}$	$E_{4,4}$	0.56	−0.512	0.157	0.579	−0.005	−0.014	−0.141	0.588	0.276	−0.022	0.89

Table 6
Semi-axis lengths derived by the proposed method for the HSV dataset.

Rule #	Unions of ellipses	$E_{t,k}$	$b_{t,k,1}$	$b_{t,k,2}$	$b_{t,k,3}$	$b_{t,k,4}$	$b_{t,k,5}$	$b_{t,k,6}$	$b_{t,k,7}$	$b_{t,k,8}$	$b_{t,k,9}$	$b_{t,k,10}$	$b_{t,k,11}$
R_1	$E_{1,1}\cup\ldots\cup E_{1,7}$	$E_{1,1}$	4.957	2.241	0.647	0.266	10.745	0.154	0.599	36 264 300	0.544	0.301	26 213 090
		$E_{1,2}$	31.807	0.427	0.069	1 563 767 000	0.156	15.347	23.008	0.488	16.045	2.481	22 934.65
		$E_{1,3}$	82.359	3620.245	138.5	279.576	89 354.84	182.989	542.556	2318.924	69.17	16.145	2.684
		$E_{1,4}$	233.282	107.934	165.786	5.284	8.045	7160.408	315.22	3.892	24.955	54.222	9.158
		$E_{1,5}$	218.398	379.603	7.813	17.026	11.019	22.493	20.919	39.983	0.353	2.427	3.989
		$E_{1,6}$	1.83	127.361	2.068	3.9	8.599	143 538.8	9.185	106.733	1.473	0.058	1.918
		$E_{1,7}$	0.981	1.558	1.94	1.341	1.041	1.065	1.589	1.446	2.144	1.523	1.744
R_2	$E_{2,1}\cup E_{2,2}\cup E_{2,3}$	$E_{2,1}$	14.996	13.386	5.181	39.177	2 642 085 000	3.596	0.528	2052.08	5.312	0.26	1.196
		$E_{2,2}$	210.784	0.949	93.93	4.694	1.134	6.301	8.297	1.524	8.749	0.194	3.477
		$E_{2,3}$	1.366	1.57	1.466	1.7	1.388	1.638	1.553	1.605	1.446	1.548	1.617
R_3	$E_{3,1}$	$E_{3,1}$	432.372	149.685	0.712	159.744	6.993	2.343	7.495	87.633	0.684	0.89	0.189
	$E_{3,2}$	$E_{3,2}$	1.251	1.23	1.23	1.371	1.171	0.884	1.666	1.211	1.497	1.581	0.148
R_4	$E_{4,1}$	$E_{4,1}$	237.942	23.132	167.056	4342.534	5049.663	22 386.67	21 218.82	3254.544	15 977.12	7.016	4925.845
	$E_{4,2}$	$E_{4,2}$	120.127	645.265	2.677	3.736	0.682	5.003	11.027	7.281	3.128	0.255	20.025
	$E_{4,3}$	$E_{4,3}$	62.618	6.207	464.303	290.558	0.732	33.94	1.797	83.788	15.567	1338.22	486.132
	$E_{4,4}$	$E_{4,4}$	1.266	2.013	1.454	1.722	1.787	1.288	1.736	1.912	1.614	4.186	1.956

The decision tree-based method in Table 2 induces rules for the same dataset. For example, the following 6 branch lines indicate that $AR(R_1, R_2, R_3) = (1, 0.98, 0.98)$, $SR(R_1, R_2, R_3) = (1, 0.98, 0.98)$, and $CR(R_1, R_2, R_3) = 3/6 = 0.5$.

- R_1: if $(a_{i,3} < 3)$ then objects i belong to class 1.
- R_2: if $(a_{i,3} \geq 3) \cap (a_{i,3} < 5) \cap (a_{i,4} < 1.7)$ or if $(a_{i,3} \geq 5) \cap (a_{i,4} < 1.8)$ $(a_{i,4} \geq 1.6)$ then objects i belong to class 2.
- R_3: if $(a_{i,3} \geq 3) \cap (a_{i,3} < 5) \cap (a_{i,4} < 1.8) \cap (a_{i,4} \geq 1.7)$ or if $(a_{i,3} \geq 5)$ $\cap(a_{i,4} < 1.6)$ or if $(a_{i,3} \geq 3) \cap (a_{i,4} \geq 1.8)$ then objects i belong to class 3.

The hyper sphere method was reported by Huang et al. [14], whose method has two hyper spheres and one union hyper sphere. The report indicates $AR(R1, R2, R3) = (1, 1, 1)$, $SR(R1, R2, R3) = (1, 0.98, 0.98)$ and $CR(R1, R2, R3) = 3/3 = 1$.

These methods demonstrate excellent performance in terms of accuracy, support and compactness. The proposed method outperforms the other two methods.

4.2. Swallow dataset

The European barn swallow (*Hirundo rustica*) dataset was obtained by trapping individual swallows in Stirlingshire, Scotland, between May and July 1997. The dataset contains 69 objects (i.e., swallows), each of which is described by eight attributes and classified according to its gender.

The dataset is solved by the linearized hyper ellipse model with 32 piecewise line segments to induce the classification rules. Table 3 lists the optimal solutions (i.e., centroid points and semi-axis lengths) for rules R_1 and R_2. The performance of the proposed method in inducing classification rules is compared with that of the decision tree-based [20] and hyper sphere method [14]. The comparison result is listed in Table 4.

4.3. HSV dataset

The HSV dataset [29] contains information on 122 patients divided into four classes. Each patient has 11 pre-operative attributes. To maximize the support rate, the linearized hyper ellipse model is solved with 32 piecewise line segments, which generates six independent hyper ellipses and two unions of hyper ellipses. The centroid points and semi-axis lengths of the hyper ellipses are shown in Tables 5 and 6, respectively. The decision tree method generates 24 rules on the HSV dataset, and the hyper sphere method uses 7 hyper spheres and three unions of hyper spheres. The details are listed in [14]. Table 7 compares the rates (i.e., *AR*, *SR*, and *CR*) derived by the compared methods. The proposed method achieves the best performance.

4.4. Limitation of the hyper ellipse model

The hyper ellipse model is solved by the most powerful mixed-integer program software CPLEX [18]. An optimization technique was applied by using branch and bound solver. The experimental

Table 7
Comparison of the results derived by the three methods on the HSV dataset (R_1, R_2, R_3, R_4).

Items	Proposed method	Decision tree	Hyper sphere method
$AR(R_1, R_2, R_3, R_4)$	(1, 1, 1, 1)	(0.93, 0.81, 0.7, 0.71)	(1, 1, 1, 1)
$SR(R_1, R_2, R_3, R_4)$	(1, 1, 1, 1)	(0.93, 0.72, 0.78, 0.71)	(0.9, 1, 1, 1)
CR	0.5	0.17	0.4

Table 8
Different piecewise line segment of the Iris flower data set (R_1, R_2, R_3).

Items	No. of piecewise line segments		
	8	16	32
$AR(R_1, R_2, R_3)$	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)
$SR(R_1, R_2, R_3)$	(1, 0.8, 0.8)	(1, 0.9, 1)	(1, 1, 1)
CR	0.6	0.9	1
Solving time (min)	1	2	5

Table 9
Different piecewise line segment on the Swallow data set (R_1, R_2).

Items	No. of piecewise line segments		
	8	16	32
$AR(R1, R2)$	(1, 1)	(1, 1)	(1, 1)
$SR(R1, R2)$	(0.7, 0.8)	(0.8, 0.9)	(1, 1)
CR	0.5	0.8	1
Solving time (min)	0.9	1.5	4.4

Table 10
Different piecewise line segment on the HSV dataset (R_1, R_2, R_3, R_4).

Items	No. of piecewise line segments		
	8	16	32
$AR(R_1, R_2, R_3, R_4)$	(1, 1, 1, 1)	(1, 1, 1, 1)	(1, 1, 1, 1)
$SR(R_1, R_2, R_3, R_4)$	(0.8, 0.7, 0.9, 0.8)	(0.9, 0.9, 1, 1)	(1, 1, 1, 1)
CR	0.3	0.4	0.5
Solving time (min)	0.8	1.9	4.8

results reported in the Sections 4.1–4.3 illustrate that the usefulness of the proposed method is better than that of the current methods, including the decision tree method and the hyper sphere method. Solving the hyper ellipse model, which is reformulated by linearization technique using 32 piecewise line segments, takes approximately 5 minutes for each dataset that runs on CPLEX (i.e., Sections 4.1–4.3).

By surveying the piecewise linearization technique in the three datasets, Tables 8–10 show the experimental results of solving time with different number of line segments used in the linearization process by the presented method. More break points can derive an approximately global solution with higher rates of support and compactness by linearly approximating the nonlinear terms. The presented method requires much longer computation time, especially when the number of piecewise line segments is increased. The rates of support and compactness also become higher.

The present method requires extra numbers of binary variables and constraints to reformulate the approximate piecewise line segments. The computation time for solving a linearized hyper ellipse program increases rapidly as the numbers of line segments increase. The computation time of the proposed method is also slower than that of the decision tree and the hyper plane method, especially for large datasets or a great number of piecewise line segments. Maybe a mainframe-version optimization software [27,30,32] integrating meta-heuristic algorithms or distributed computing techniques can enhance the speed of solving this program.

5. Conclusions

This study proposes a novel method for inducing rules to classify objects with various classes. In solving a mathematical program, the proposed method generates a set of hyper ellipses to classify objects of the same class. The solutions are approximated to optimum because of linearization techniques. The proposed method also obtains high values in terms of accuracy rate (AR), support rate (SR), and compact rate (CR), which are better than the current methods. On the other hand, the proposed method is guaranteed to find an optimal rule, but the computational complexity grows rapidly as the problem size increases. More investigation and research are required to further enhance the computational efficiency of globally solving large-scale classification problems such as the solutions run in mainframe-version optimization software, integrate meta-heuristic algorithms, or use distributed computing techniques.

Conflict of interest statement

None declared.

Acknowledgements

The author would like to thank the editor and two anonymous referees for providing me the most valuable comments to improve the quality of this manuscript. The authors would like to thank the National Science Council, Taiwan, R.O.C. for financially supporting this research under Contract No. 102-2811-E-009-017-.

References

[1] S. Aja-Fernandez, R. de Luis-Garcia, M.A. Martin-Fernandez, C. Alberola-Lopez, A computational TW3 classifier for skeletal maturity assessment-a computing with words approach, J. Biomed. Inf. 37 (2) (2004) 99–107.
[2] D.A. Babayev, Piece-wise linear approximation of functions of two variables, J. Heuristics 2 (4) (1997) 313–320.
[3] E.M.L. Beale, J.J.H. Forrest, Global optimization using special ordered sets, Math. Program. 10 (1) (1976) 52–69.
[4] M.J. Beynon, K.L. Buchanan, An illustration of variable precision rough set theory: the gender classification of the European barn swallow (*Hirundo rustica*), Bull. Math. Biol. 65 (5) (2003) 835–858.
[5] L. Breiman, J. Friedman, R. Olshen, C. Stone, Classification and Regression Trees, Wadsworth International, Belmont, CA, 1984.
[6] C.C. Chang, C.J. Lin. LIBSVM: A Library for Support Vector Machines ⟨http://www.csie.ntu.edu.tw/ cjlin/libsvm/index.html⟩, 2010.
[7] J.X. Che, Peer-estimation for multiple criteria ABC inventory classification, Comput. Oper. Res. 38 (12) (2011) 1784–1791.
[8] Z. Chen, J. Li, L. Wei, A multiple kernel support vector machine scheme for feature selection and rule extraction from gene expression data of cancer tissue, Artif. Intell. Med. 41 (2007) 161–175.
[9] C.W. Chu, G.S. Liang, C.T. Liao, Controlling inventory by combining ABC analysis and fuzzy classification, Comput. Ind. Eng. 55 (2008) 841–851.
[10] D.C. Dunn, W.E.G. Thomas, J.O. Hunter, An evaluation of highly selective vagotomy in the treatment of chronic ulcer, Surg. Gynecol. Obstet. 150 (1980) 145–151.
[11] R.A. Fisher, The use of multiple measurements in taxonomic problems, Ann. Eugen. 7 (1936) 179–188.
[12] P. Geurts, M. Fillt, D. Seny, M.A. Meuwis, M. Malaise, M.P. Merville, L. Wehenkel, Proteomic mass spectra classification using decision tree based ensemble methods, Bioinformatics 11 (2005) 3138–3145.
[13] M. Gomathi, P. Thangaraj, A computer aided diagnosis system for lung cancer detection using machine learning technique, Eur. J. Sci. Res. 51 (2) (2011) 260–275.
[14] Y.H. Huang, Y.C. Ko, H.C. Lu, An Optimal Classification Method for Biological and Medical Data, Math. Probl. Eng. (2012), http://dx.doi.org/10.1155/2012/398232.
[15] J.T. Hwang, A study on improving training speed in support vector machine, in: IEEE 18th International Conference on Geoinformatics, 2010, pp. 1–6.
[16] IBM/ILOG. CPLEX 12.0 Reference Manual. Software Available at ⟨http://www.ilog.com/products/cplex/⟩, 2010.
[17] S. Katagiri, S. Abe, Incremental training of support vector machines using hyperspheres, Pattern Recognition Lett. 27 (2006) 1495–1504.
[18] H. Kim, W.Y. Loh, Classification trees with unbiased multiway splits, J. Am. Stat. Assoc. 96 (2001) 589–604.
[19] H.L. Li, C.J. Fu, A Linear programming approach for identifying a consensus sequence on DNA sequences, Bioinformatics 21 (2005) 1838–1845.
[20] H.L. Li, M.H. Chen, Induction of multiple criteria optimal classification rules for biological and medical data, Comput. Biol. Med. 38 (1) (2008) 42–52.

[21] H. Nunez, C. Angulo, A. Catala, Rule-extraction from support vector machines, in: Proceedings of the European Symposium on Artificial Neural Networks, 2002, pp. 107–112.
[22] J.P. Pedroso, N. Murata, Support vector machines with different norms: motivation, formulations and results, Pattern Recognition Lett. 22 (12) (2001) 1263–1272.
[23] D. Pomorski, P.B. Perchie, Inductive learning of decision trees: application to fault isolation of an induction motor, Eng. Appl. Artif. Intell. 14 (2001) 155–166.
[24] J.R. Quinlan, Induction of decision trees, Mach. Learn. 1 (1986) 86–106.
[25] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufman, San Mateo, CA, 1993.
[26] R.M. Rifkin, Everything Old is New Again: A Fresh Look at Historical Approaches in Machine Learning, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, 2002.
[27] M. Rockville, Large Scale Computing and Storage Requirements for Biological and Environmental Research, Technical Report, Ernest Orlando Lawrence Berkeley National Laboratory, Berkeley, CA, USA, 2009.
[28] V.N. Vapnik, The Nature of Statistical Learning Thoery, Springer-Verlag, New York, 1995.
[29] K. Slowinski, Rough classification of HSV patients, in: R. Slowinski (Ed.), Intelligent Decision Support-Handbook of Applications and Advances of the Rough Sets Theory, Kluwer, Dordrecht, Netherlands, 1992, pp. 77–94.
[30] C. Than, R. Sugino, H. Innan, L. Nakhleh, Efficient inference of bacterial strain trees from genome-scale multi-locus data, Bioinformatics 24 (2008) i123–i131.
[31] M.H. Tsai, Y.R. Yeh, Y.J. Lee, Y.C.F. Wang, Solving nonlinear SVM in linear time? A Nystrom approximated SVM with applications to image classification, in: The 13th IAPR Conference on Machine Vision Applications (MVA), 2013.
[32] X. Xie, X. Fang, S. Hu, D. Wu, Evolution of supercomputers, Front. Comput. Sci. China 4 (4) (2010) 428–436.
[33] H. Zhang, C.Y. Yu, B. Singer, M. Xiong, Recursive partitioning for tumor classification with gene expression microarray data, Proc. Natl. Acad. Sci. USA 98 (2001) 6730–6735.
[34] M.L. Zhu, S.F. Chen, X.D. Liu, Sphere-structured support vector machines for multi-class pattern recognition, Lect. Notes Comput. Sci. 2639 (2003) 589–593.