Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2014, Article ID 318478, 9 pages
http://dx.doi.org/10.1155/2014/318478

Research Article

A New Method for Solving Supervised Data Classification Problems

Parvaneh Shabanzadeh[1,2] and Rubiyah Yusof[1,2]

[1] *Centre for Artificial Intelligence and Robotics, Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia*
[2] *Malaysia-Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia*

Correspondence should be addressed to Parvaneh Shabanzadeh; parvaneh.shabanzade@gmail.com
and Rubiyah Yusof; rubiyah.kl@utm.my

Received 14 July 2014; Revised 19 October 2014; Accepted 6 November 2014; Published 27 November 2014

Academic Editor: Luiz Duarte

Supervised data classification is one of the techniques used to extract nontrivial information from data. Classification is a widely used technique in various fields, including data mining, industry, medicine, science, and law. This paper considers a new algorithm for supervised data classification problems associated with the cluster analysis. The mathematical formulations for this algorithm are based on nonsmooth, nonconvex optimization. A new algorithm for solving this optimization problem is utilized. The new algorithm uses a derivative-free technique, with robustness and efficiency. To improve classification performance and efficiency in generating classification model, a new feature selection algorithm based on techniques of convex programming is suggested. Proposed methods are tested on real-world datasets. Results of numerical experiments have been presented which demonstrate the effectiveness of the proposed algorithms.

1. Introduction

Supervised data classification is a widely used technique in various fields, including data mining, whose aim is to establish rules for the classification of some observations assuming that the classes of data are known. Due to the explosive growth of both business and scientific databases, extracting efficient classification rules from such databases is of major importance.

In the past 30 decades various algorithms were designed for supervised data classification which are based on completely different approaches, for example, statistics methods [1], neural networks [2], genetic algorithms [3], graphical models [4], and adaptive spline methods [5].

Algorithms based on inductive logic programming [6] and hybrid systems [7] are also used for supervised data classification. Kotsiantis in 2007 and Mangasarian and Musicant in 2001 [8, 9] presented good review of these approaches, including their definition and comparison. One of the new and most promising approaches to supervised data classification is based on methods of mathematical optimization. There exist different ways for the application of optimization; for example, see [10–12]. One of these methods is based on finding clusters for the given training sets. The data vectors are allocated to the closest cluster and correspondingly to the set, which contains this cluster [13].

On the other hand, one of the most influencing important factors on the classification accuracy rate is feature selection. If the dataset contains a number of features, the dimension of the space will be large and nonclean, degrading the classification accuracy rate [14]. An efficient and robust feature selection method can eliminate noisy, irrelevant, and redundant data [15]. Therefore reducing it without loss of useful information is expected to accelerate the algorithms and increase the accuracy. Most feature selections are based on statistical considerations, and the features are usually removed according to a correlation between observations and features (see [15, 16]). In [17, 18] approaches based on optimization techniques have been developed. Then a new feature selection algorithm based on techniques of convex programming is proposed.

So in this research, new algorithms for classification and feature selection problems based on optimization techniques are designed; for the execution of these approaches one needs to solve complex problems of nonconvex and nonsmooth unconstrained optimization, either local or global. Despite the nonsmoothness and nonconvexity of the objective functions, global methods are much simpler and more applicable than local ones. In the present research, we are adapted and used as one type of direct global optimization methods, namely, mesh adaptive direct search (MADS) [19]. MADS is derivative-free method in the sense that this method does not compute nor even attempt to evaluate derivatives. Mesh adaptive direct search methods are designed to only use function values and require only a numerical value of the objective; no knowledge about the internal structure of the problem is needed [19].

Results of computational experiments using real-world datasets were presented and compared with the best known solutions from the literature.

The paper is organized as follows. The optimization approach to classification is considered in Section 2. In Section 3, an algorithm for feature selection problems is studied. In Section 4 an algorithm is presented for solving optimization problems. The discussion of the results of computational experiments and their analysis is explained in Section 5. Finally Section 6 concludes the paper.

2. A New Optimization Algorithm for Solving Classification Problem

Consider a set D consisting of m points $d^1, \ldots, d^m$ which contains k classes, that is, k nonempty finite subsets D_j, $j = 1, \ldots, k$ of n-dimensional space R^n. Assume that the set D_j consists of e_j points ($j = 1, \ldots, k$). The aim of classification is to categorize a new observation into one of the known classes and there are many existing approaches for solving this problem (as mentioned in Introduction). In continue, a classification method that is based on optimization ways has been studied. Numerical experiments verify that this method outperforms known ones for real-world databases. In order to solve this problem, the clusters of each class of dataset have to be identified, which must be done along with centers of the corresponding clusters. New observations are allocated to the class with least distance between its centers.

Thus, at the first finding the clusters of a finite set will be explained. Clustering in n-dimensional Euclidean space R^n is based on some similarity (distance) metric, the Minkowski metric was used for this aim. There are various methods for solving clustering problem. One of the most popular methods is the center based clustering model [20–22].

Consider the set D; suppose that this set consists of only one cluster; thus its center can be calculated by solving the following convex programming problem:

$$\begin{aligned} \text{Min} \quad & f_1(y) = \sum_{i=1}^{m} \left\| y - d^i \right\| \\ \text{s.t.} \quad & y \in R^n. \end{aligned} \tag{1}$$

Suppose that y_1 is the solution of problem (1); in order to find a center of the second cluster, find the answer of the following optimization problem:

$$\begin{aligned} \text{Min} \quad & f_2(y) = \sum_{i=1}^{m} \min \left\{ \left\| y^1 - d^i \right\|, \left\| y - d^i \right\| \right\} \\ \text{s.t.} \quad & y \in R^n. \end{aligned} \tag{2}$$

In the same manner, suppose the already calculated $(h-1)$ centers, and then the center y^h of hth cluster is described as a solution to the following problem:

$$\begin{aligned} \text{Min} \quad & f_h(y) = \sum_{i=1}^{m} \min \left\{ \left\| y^1 - d^i \right\|, \left\| y^2 - d^i \right\|, \ldots, \right. \\ & \qquad \left. \left\| y^{h-1} - d^i \right\|, \left\| y - d^i \right\| \right\} \\ \text{s.t.} \quad & y \in R^n. \end{aligned} \tag{3}$$

Then the following algorithm for solving a classification problem is proposed. Suppose that database contains 2 classes: D_1 and D_2. Let $M_1 = \{1, \ldots, |D_1|\}$, $M_2 = \{1, \ldots, |D_2|\}$, and $\epsilon > 0$ a tolerance.

Algorithm 1. A new algorithm for classification problem is presented.

Step 1 (initialization). Suppose that sets B_1 and B_2 contain a unique cluster; calculate the centers of clusters by solving the following problems:

$$\begin{gathered} \text{Min} \sum_{\mathbf{i} \in \mathbf{M}_1} \left\| \mathbf{y}^1 - \mathbf{d}^\mathbf{i} \right\| \\ \text{Min} \sum_{\mathbf{i} \in \mathbf{M}_2} \left\| \mathbf{y}^2 - \mathbf{d}^\mathbf{i} \right\|. \end{gathered} \tag{4}$$

Suppose that $\mathbf{y}_{11}^*$, $\mathbf{y}_{21}^*$ are the solutions to these problems and allow $\mathbf{f}_{11}^*$ and $\mathbf{f}_{21}^*$ to be the values of these problems, respectively. Let $h = 1$.

Step 2 (identify the sets of points "misclassified" by the current clusters). Compute the sets

$$\begin{gathered} M_{1h}^* = \left\{ i \in M_1 : \min_{r=1,\ldots,h} \left\| y_{2r}^* - d^i \right\| \leq \min_{r=1,\ldots,h} \left\| y_{1r}^* - d^i \right\| \right\} \\ M_{2h}^* = \left\{ i \in M_2 : \min_{r=1,\ldots,h} \left\| y_{1r}^* - d^i \right\| \leq \min_{r=1,\ldots,h} \left\| y_{2r}^* - d^i \right\| \right\}. \end{gathered} \tag{5}$$

Step 3. If $h \neq 1$, compute the following sets:

$$\begin{gathered} S_1 = \left\{ i \in M_1 \setminus M_{1h}^* : \left\| y_{1h}^* - d^i \right\| \leq \min_{r=1,\ldots,h-1} \left\| y_{1r}^* - d^i \right\| \right\} \\ S_2 = \left\{ i \in M_2 \setminus M_{2h}^* : \left\| y_{2h}^* - d^i \right\| \leq \min_{r=1,\ldots,h-1} \left\| y_{2r}^* - d^i \right\| \right\}. \end{gathered} \tag{6}$$

Else

$$S_1 = \{i \in M_1 \setminus M_{1h}^*\}$$
$$S_2 = \{i \in M_2 \setminus M_{2h}^*\}. \tag{7}$$

Step 4. Improve the center of the cluster by solving the following convex programming problems:

$$\text{Min} \quad \sum_{i \in S_1} \left\| y^1 - d^i \right\| \tag{8}$$

$$\text{Min} \quad \sum_{i \in S_2} \left\| y^2 - d^i \right\| \tag{9}$$

s.t. $y^j \in R^n$, $j = 1, 2$.

Allow y^{01} and y^{02} to be the solutions of the problems (8) and (9), respectively. Set $y_{1h}^* = y^{01}$ and $y_{2h}^* = y^{02}$.

Step 5 (checking the stopping criterion). If $h \neq 1$, calculate these functions:

$$f_{1h} = \sum_{i \in M_1} \min \left\{ \left\| y_{1h}^* - d^i \right\|, \left\| y_{11}^* - d^i \right\|, \ldots, \left\| y_{1,h-1}^* - d^i \right\| \right\}$$
$$f_{2h} = \sum_{i \in M_2} \min \left\{ \left\| y_{2h}^* - d^i \right\|, \left\| y_{21}^* - d^i \right\|, \ldots, \left\| y_{2,h-1}^* - d^i \right\| \right\} \tag{10}$$

and if $\{|f_{1h} - f_{1,h-1}|/f_{11}, |f_{2h} - f_{2,h-1}|/f_{21}\} < \varepsilon$, then the algorithm ends. Otherwise go to Step 6.

Step 6 (determine the estimate of next cluster). Solve the following optimization problems:

$$\text{Min} \sum_{i \in M_1} \min \left\{ \left\| y^1 - d^i \right\|, \left\| y_{11}^* - d^i \right\|, \ldots, \left\| y_{1h}^* - d^i \right\| \right\} \tag{11}$$

$$\text{Min} \sum_{i \in M_2} \min \left\{ \left\| y^2 - d^i \right\|, \left\| y_{21}^* - d^i \right\|, \ldots, \left\| y_{2h}^* - d^i \right\| \right\} \tag{12}$$

s.t. $y^j \in R^n$, $j = 1, 2$.

Step 7. Allow y^{11} and y^{21} to be the solutions of the problems (11) and (12), respectively. Set $y_{1,h+1}^* = y^{11}$, $y_{2,h+1}^* = y^{21}$, and $h = h + 1$ and go to Step 2.

3. Feature Selection Algorithm

Feature selection is concerned with the identification of a subset of features that significantly contributes to the discrimination or prediction problem. The main goal of feature selection is to search for an optimal feature subset from the initial feature set that leads to improved classification performance and efficiency in generating classification model. During the past decades, wide research has been conducted by researchers from multidisciplinary fields including data mining, pattern recognition, statistics, and machine learning. In [23] a comparison of various feature selection algorithms for large datasets is presented.

Consider a database which contains 2 nonempty finite sets $B_j \subset R^m$, $j = 1, 2$. Let $M_j = \{1, \ldots, |B_j|\}$, $j = 1, 2$, where $|B|$ denotes the cardinality of a finite set B. Let $T_j \epsilon \{1, 2, \ldots\}$, $j = 1, 2, 3$ be the thresholds and let $\varepsilon > 0$ be some tolerance.

Algorithm 2. Feature selection.

Step 1 (initialization). Set $t = 1, I_t = \{1, \ldots, m\}$.

Step 2. Find centers of clusters by assuming that the sets $B_j, j = 1, 2$ contain a unique cluster. Compute the centers of clusters by solving the following problems of convex programming:

$$\text{Minimize} \quad \sum_{i \in M_j} \left\| x^j - b^{ij} \right\|_q$$
$$\text{subject to} \quad x^j \epsilon R^m, \quad j = 1, 2. \tag{13}$$

Here $\|x\|_q$ is defined by $\|x\|_q = (\sum_{t=1}^m |x_t|^q)^{1/q}$.

Step 3. Find points of the set $B_j, j = 1, 2$ which are closer to the cluster center of the other set (bad points).

Let $x_*^j, j = 1, 2$ be solutions to (13). Compute the sets

$$M_1^t = \left\{ i \epsilon M_j : \left\| x_*^2 - b^{i1} \right\|_q \leq \left\| x_*^1 - b^{i1} \right\|_q \right\}$$
$$M_2^t = \left\{ i \epsilon M_2 : \left\| x_*^1 - b^{i2} \right\|_q \leq \left\| x_*^2 - b^{i2} \right\|_q \right\}. \tag{14}$$

Set $M_3^t = M_1^t \cup M_2^t$.

If $t = 0$, then go to Step 5; otherwise go to Step 4.

Step 4. Calculate

$$EN_i = \left| M_i^t - M_i^{t-1} \right|, \quad i = 1, 2, 3$$
$$ff_i = EN_i - T_i, \quad i = 1, 2, 3 \tag{15}$$
$$If \quad \max \{ff_i, i = 1, 2, 3\} > 0.$$

Then I_{t-1} is a subset of most informative attributes and the algorithm terminates. Otherwise go to Step 5.

Step 5. To determine the closest coordinates, calculate

$$D_0 = \min \left\{ \left| \left(x_*^1\right)_r - \left(x_*^2\right)_r \right| : r \epsilon I_t \right\} \tag{16}$$

and define the following set:

$$R_t = \left\{ r \epsilon I_t : \left| \left(x_*^1\right)_r - \left(x_*^2\right)_r \right| \leq D_0 + \epsilon \right\}. \tag{17}$$

Step 6. Construct the set

$$I_{t+1} = I_t \setminus R_t. \tag{18}$$

If $I_{t+1} = \emptyset$, then I_t is the subset of most informative attributes. If $|I_{t+1}| = 1$,then I_{t+1} is the subset of most informative attributes. Then the algorithm terminates; otherwise set $t = t + 1$ and go to Step 2.

4. Solving Optimization Problems

In this section, algorithm for solving problems as mentioned in the classification algorithm has been discussed. Since these functions are nonsmooth and estimate of subgradients is difficult, direct search methods of optimization seem to be the best option for solving them. The main attraction of direct search methods is their ability to find optimal solutions without the need for computing derivatives, in contrast to the more familiar gradient-based methods [24].

Direct search algorithms can be applied for problems that are difficult to be solved with traditional optimization techniques, including problems that are difficult to model mathematically or are not well defined. They can be also applied when the objective function is discontinuous, stochastic, highly nonlinear, or undefined derivative.

In general, direct search algorithms are called pattern search algorithms and both the generalized pattern search (GPS) algorithm and the MADS algorithm are pattern search algorithms that compute a sequence of points that get closer and closer to the optimal point. At each step, the algorithm investigates a set of points, called a mesh, around the current point (the point computed at the previous step of the algorithm). The mesh is created by adding the current point to a scalar multiple of a set of vectors called a pattern. If the pattern search algorithm discovers a point in the mesh that makes better (decreases) the objective function at the current point, the new point becomes the current point at the next step of the algorithm.

4.1. The MADS Method. MADS methods are designed to only use function values and require only a numerical value of the objective; no knowledge about the internal structure of the problem is needed. These methods can quickly and easily be used in nonlinear, nonconvex, nondifferentiable, discontinuous, or undermined problems [19]. The convergence analysis of MADS guarantees necessary optimality conditions of the first and second orders under certain assumptions [19]. A general optimization problem can be as follows:

$$\begin{aligned} \text{Min} \quad & F(x) \\ \text{s.t.} \quad & x \in X, \end{aligned} \tag{19}$$

where $F : R \rightarrow R^n \cup \{+\infty\}$, $X = \{x \in R^n \mid C_i(x) \leq 0, i = 1, \ldots, m, L_1 \leq x \leq L_2\}$, $C : R^n \rightarrow R^m$, and $L_1 \in (\{-\infty\} \cup R)^n$, $L_2 \in (\{+\infty\} \cup R)^n$.

MADS is an iterative algorithm. Each iteration (shown by the subscript k) is initiated with the current best feasible solution x_k, known as the incumbent solution, and each iteration k of the MADS algorithm can be stated by two steps. First, an optional search step over the space of variables is performed as long as it is a finite process and all trial points lie on a mesh. If no better point is found or no global search is applied, the algorithm goes to a compulsory local exploration step (compulsory because it ensures convergence). Second is the poll step; at most $2n$ trial mesh points near the incumbent solution are chosen (the poll set) and evaluated. If no better neighbor is found, the mesh is refined. If an improved mesh point $x_{k+1} \epsilon X$ is found, the mesh is kept the same or coarsened, and then x_{k+1} is the next incumbent. The exploration directions vary at each iteration and become dense with probability 1. This is the main difference between the pattern search and MADS algorithms. General constraints can be handled with a barrier approach, which redefines the objective as in the following equation:

$$F_X = \begin{cases} F(x) & \text{if } x \in X \\ +\infty & \text{otherwise.} \end{cases} \tag{20}$$

Then, MADS is applied to the unconstrained barrier problem

$$\text{Min}_x \ F_X(x). \tag{21}$$

The feasible region X can be nonlinear, nonconvex, nondifferentiable, or disjoint. There are no hypotheses made on the domain, except that the initial point must be feasible. The convergence results depend on the local smoothness of F (and not F_X, which is obviously discontinuous on the boundary of X).

Algorithm 3 (the MADS algorithm). A general and flexible algorithmic framework for MADS is studied in [19]. This general framework is then specialized to a specific algorithmic implementation. The main steps of the algorithm are summarized as follows.

Step 1 (initialization). The user defines the starting point and the initial mesh size.

The algorithm initializes other parameters for subsequent steps.

Step 2 (request for an improved mesh point). Consider the following steps:

(i) global search (optional): evaluation of F over a finite subset of points defined by the mesh;

(ii) local poll (mandatory): definition of a poll set and evaluation of F over points in that set.

Step 3 (parameters update). Parameters are updated.

Step 4 (termination). If some stopping criterion is reached, stop; if not, go back to Step 2.

5. Results of Numerical Experiments

To verify the efficiency of the proposed algorithms a number of numerical experiments with real-world data sets have been carried out on a PC, Intel Core 2 Duo CPU, 1.95 GB of RAM.

The Australian credit dataset, the breast cancer dataset, the diabetes dataset, the heart disease dataset, the liver-disorder dataset, the German Numer dataset, and the mushroom dataset have been applied in numerical experiments.

The description of these datasets can be found in UCI Machine Learning Repository [25].

In Table 1, Nshows the number of samples of dataset, C presents the number of classes of dataset, and F is the number of features.

Table 1: Properties of the examined databases.

Data	N	C	F
Australian credit	690	2	14
Breast cancer	569	2	30
Diabetes	768	2	8
Heart	303	2	13
Liver	345	2	6
German Numer	1,000	2	24
Mushrooms	8,124	2	112

First, all features were normalized. This is done by a nonsingular matrix so that standard deviation values of all features are 1. In order to evaluate the performance, 10-fold cross-validation was used where a sample from each dataset was selected and then divided into 10 equal sized subsets. Next, a subset was selected and designated as the test set and the union of the remainder nine subsets was used as the training set. After the application of Algorithm 2 which calculates the subset of informative attributes and selection of the features, the classification model was validated with the test subset. This process was repeated where each of the 10 subsets was successively selected as the test set. Accordingly, the proposed method was run 10 times and the classification accuracy rate was calculated by averaging across all 10 test runs.

Note. In feature selection algorithm (Algorithm 2) T_i, $i = 1, 2, 3$ (maximum numbers of added "bad points" in each iteration of the feature selection algorithm for each class of dataset) have important role in the execution of this algorithm and therefore in numerical experiments one or two percent of value of each class dataset for T_1, T_2, and $T_3 = T_1 + T_2$ have been considered

In comparison with T_i, $i = 1, 2, 3$ introduced in [18] results of numerical experiments have shown that this algorithm significantly reduces the number of attributes, so that 3 attributes were used in the diabetes dataset, the breast cancer dataset, the liver-disorder dataset, and the Australian credit dataset, 11 attributes in the heart disease dataset, 4 attributes in the German dataset, and 6 attributes in the mushroom dataset for solving classification problem. While in comparison with those obtained by the proposed method and the results obtained in [18] we can see that, for the Australian credit dataset, the number of features is decreased from 6 to 3; for the breast cancer dataset it was the same, while for the heart disease dataset, the number of features is increased from 3 to 11.

In numerical experiments Algorithm 1 was used for classification of datasets with 10-fold cross-validation and the MADS algorithm has been applied for solving problems in Algorithm 1; then in this research it is called MA and is supposed as $\varepsilon = 0.01$. Results of the numerical experiments are presented in Tables 2–8. In Tables 2–8, e_{train} represents the error rate for the training data and e_{test} shows the error rate for the test data, that is, criteria for goodness of one method.

Table 2: Results in the Australian credit dataset.

Algorithm	e_{test}	e_{train}
MA	7.3	15.4
NBTree	16.8	
RBF	43.29	
KStar	19.18	
Ridor	12.65	
VFI	16.47	
MultiBoost	12.71	
Bayes net	12.13	
PSO	18.77	
Different approaches from Michie	13.1	13.2

Also the numerical results of the parametric misclassification minimization (PMM) [26], robust linear programming (RLP) [27], the hybrid misclassification minimization (HMM) [28], support vector machines algorithms [29], the K-nearest neighbor algorithm (KNN), the multilayer perceptron (MLP), the probabilistic neural network (PNN), and the sequential minimal optimization algorithm (SMO) [30, 31] were used for the purpose of comparison. Moreover the results obtained by particle swarm optimization algorithm (PSO) [32, 33], music-inspired harmony search algorithm (HS) [34], fire fly algorithm (FFA) [35] and its references, the Waikato Environment for Knowledge Analysis (WEKA) system release 3.4 [36], which contains a large number of such techniques that were divided into different groups, were equally used for comparison. From each of such groups, some representatives have been chosen. They are as follows: the radial basis function artificial neural network (RBF) [37], among the lazy, the KStar [38], among the rule-based ones the ripple down rule (Ridor) [39], and among others the voting feature interval (VFI) [40]. Similarly, we have the MultiBoostAB [41] and among the Bayesian the Bayes net [42]. Parameter values used for any technique are those set as default in WEKA. Also the results obtained by support vector machines algorithm [10], IncNet [43], fuzzy approach [44], FLEXNFIS [45], FNN [46], RULES-4 [47] and C4.5 [48], Naïve Bayes [49, 50], BNND and BNNF methods from [51], SSVM [52], RSVM [53], SVM [54], LSSVM [55], FAIRS [56], DC-RBFNN [57], Boost [58], RIPPER [59], INB [60], and GPF [61] were used in the experiments.

The results of numerical experiments obtained by using 23 algorithms of classification from Michie [62], presented in Chapter 9 of this book, were also applied; these are statistical, neural network, and machine learning algorithms. In addition, only the best results obtained by these algorithms are presented in Tables 2–8.

The results for the Australian credit database are presented in Table 2, which indicates that the accuracy of the proposed method is higher than the accuracies of other methods pointed out in the table.

The results for second database, breast cancer database, are presented in Table 3. It shows that the accuracy of proposed method is higher than the accuracies of other

Table 3: Results in the breast cancer database.

Algorithm	e_{test}	e_{train}
MA	2.5	2.9
PMM	3.5	1.4
RLP	2.8	2.3
HMM	2.6	2.1
NBTree	7.69	
RBF	20.27	
KStar	2.44	
Ridor	6.36	
VFI	7.34	
MultiBoost	5.59	
Bayes net	4.19	
PSO	3.49	

Table 4: Results in the diabetes dataset.

Algorithm	e_{test}	e_{train}
MA	19.7	18.0
PMM	23.3	19.4
RLP	24.0	23.3
HMM	24.1	21.6
SVM	25.0	24.0
NBTree	25.52	
RBF	39.16	
KStar	34.05	
Ridor	29.31	
VFI	34.37	
MultiBoost	27.08	
Bayes net	25.52	
PSO	21.77	
IncNet	22.4	
Fuzzy approach	22.4	
FLEXNFIS	21.4	
FNN	18.2	
Different approaches from Michie	22.3	

methods except for KStar and HMM methods in which the accuracies are quite close to that of the proposed method.

For the diabetes database, the results of numerical experiments are presented in Table 4, which shows that the accuracy of proposed method is higher than the accuracies of other methods pointed out in this table except that of FNN method in which the accuracy is the best.

The results for the heart database are presented in Table 5. From these results and the previous results, it is safe to conclude that the accuracy of proposed method is the best and, thus, the most suitable for this dataset.

The results for the liver database are presented in Table 6 which shows that the accuracy of proposed method is better than the accuracies of other methods pointed out in the table except for PMM method in which the accuracy is the best. From these results and the previous results, it is safe to conclude that the accuracy of proposed method is the best and, thus, the most suitable for this dataset.

Table 5: Results in the heart dataset.

Algorithm	e_{test}	e_{train}
MA	14.8	14.5
PMM	17.8	8.6
RLP	16.5	15.5
HMM	17.2	12.5
SVM	24.1	15.3
NBTree	22.36	
RBF	45.25	
KStar	26.70	
Ridor	22.89	
VFI	18.42	
MultiBoost	18.42	
Bayes net	18.42	
PSO	15.73	
Different approaches from Michie	37.4	35.1

Table 6: Results in the liver dataset.

Algorithm	e_{test}	e_{train}
MA	27.8	23.0
PMM	31.6	25.1
RLP	33.1	31.0
HMM	33.4	27.8
NBTree	39.0	39.8
RULES-4	44.1	
C4.5	34.5	
Naïve Bayes	36.6	
BNND	38.6	
BNNF	38.2	

Table 7: Results in the German Numer dataset.

Algorithm	e_{test}	e_{train}
MA	25.53	24.7
PNN	31.71	
C4.5	28.53	
SMO	25.16	
Boost	28.81	
Bayes	25.66	
DC-RBFNN	25.29	
MLP	27.86	
KNN	31.52	
RIPPER	30.0	
GPF	24.76	
FFA	46.59	
HS	44.76	
PSO	40.48	

The results for the German database are presented in Table 7, which confirms that the errors of the proposed method are lower than the errors of other methods pointed out in the table, except for SMO, DC-RBFNN, and GPF methods in which the errors are lower than MA method.

Table 8: Results in the mushroom dataset.

Algorithm	e_{test}	e_{train}
MA	0.27	0.26
PNN	0.29	
C4.5	0	
SMO	0	
Boost	3.21	
Bayes	4.46	
DC-RBFNN	1.67	1.77
MLP	0	
KNN	0.17	
INB	4.7	
RIPPER	0	
FFA	0	
HS	0.05	
PSO	0.04	

The results for the last database, mushroom database, are presented in Table 8. It shows that the errors of the proposed method are near 0. These are showing goodness of the proposed method.

As shown in Tables 2–8, the MA model obtains the best or near the best prediction accuracies in almost all datasets.

Further, in order to evaluate important factors in the performance of the MADS algorithm for solving the classification problem, different experiments were carried out on the datasets as mentioned earlier in this paper. Here only the main results obtained are presented from different experiments conducted; this is done to avoid unnecessary details for the sake of summary. In this research, mesh factors ($\Delta^r_{\text{ME}}, \Delta^r_{\text{MC}}$) have been defined as: Δ^r_{ME}, is mesh contraction factor used when iteration is unsuccessful, Δ^r_{ME} is mesh expansion factor which expands mesh when iteration is successful and $\Delta^r_{\text{ME}} = 1/2^r, \Delta^r_{\text{ME}} = 2^r, r \in N$. Also it was found that the e_{test} decreases when r increases and the best value for r is near 5.

Since the direction of poll set is chosen as random in MADS algorithm, therefore each performance of this algorithm gives new result and so MA method was performed 10 times and average of solutions is presented in the following tables. Also it was found that the standard deviations of them (solutions) are near zero.

Various experiments have been accomplished on the datasets as mentioned before in the classification algorithm with having different values of ε (so that $\varepsilon \epsilon [1, 10^{-2}]$) for finding the best value for ε. Therefore, the good value was observed around 10^{-2}. Also to appraise important factors in the performance of the MADS algorithm for solving classification problem, the same experiments have been done by different strategies have been made in the MADS algorithm for step search (that means (1) search step is empty, (2) when $n+1$ random direction was chosen for mesh set in search step, (3) when genetic algorithm was chosen for step search, and (4) when Nelder-Mead algorithm was chosen for step search); and the results were almost the same.

Therefore, the results are presented in Tables 2–8, show that AM gives good results compared with other methods for all datasets. The results of numerical experiments demonstrate that the proposed algorithms are effective for solving classification problems.

6. Conclusions

In this paper a new algorithm was proposed for solving classification problem where the algorithm includes the nonsmooth and nonconvex optimization problems. The new proposed algorithm is based on classes in the database which use cluster centers so that, for each class, the cluster analysis problem with more estimation is solved.

The MADS method was used for solving the nonsmooth optimization problems. The new method was tested using real-world datasets. Results of these computational experiments show the effectiveness of the new algorithms. In the future, the size of datasets will increase; obviously applying feature selection is useful for classification problem and therefore it seems that the feature selection procedure should be further studied. Also proposing new globalization strategies for this method based on combining with other good methods similar to PSO for solving classification problems as future study is suggested.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank the Ministry of Education Malaysia for funding this research project through a Research University Grant of Universiti Teknologi Malaysia (UTM), project titled "Dimension Reduction & Data Clustering for High Dimensional & Large Dataset" (04H40). Also, thanks are due to the Research Management Center (RMC) of UTM for providing an excellent research environment in which this work was completed.

References

[1] L. Breinman, R. A. Olshen, and C. J. Stone, *Classification Adregression Trees*, Wadsworth & Brooks, Pacific Grove, Calif, USA, 1984.

[2] G. P. Zhang, "Neural networks for classification: a survey," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 30, no. 4, pp. 451–462, 2000.

[3] A. Freitas, "A survey of evolutionary algorithms for datamining and knowledge discovery," in *Advances in Evolutionary Computation*, Springer, 2002.

[4] W. Buntine, "Graphical models for discovering knowledge," in *Advances in Knowledge Discovery and Data Mining*, pp. 59–82, 1996.

[5] J. H. Friedman, "Multivariate adaptive regression splines," *The Annals of Statistics*, vol. 19, no. 1, pp. 1–141, 1991.

[6] S. Dzeroski, "Inductive logic programming and knowledge discovery in databases," in *Advances in Knowledge Discovery and Data Mining*, pp. 117–152, 1996.

[7] B. Boutsinas and M. N. Vrahatis, "Artificial nonmonotonic neural networks," *Artificial Intelligence*, vol. 132, no. 1, pp. 1–38, 2001.

[8] S. B. Kotsiantis, "Supervised machine learning: a review of classification techniques," *Informatica*, vol. 31, no. 3, pp. 249–268, 2007.

[9] O. L. Mangasarian and D. R. Musicant, "Lagrangian support vector machines," *Journal of Machine Learning Research*, vol. 1, no. 3, pp. 161–177, 2001.

[10] P. S. Bradley and O. L. Mangasarian, "Massive data discrimination via linear support vector machines," *Optimization Methods and Software*, vol. 13, no. 1, pp. 1–10, 2000.

[11] O. L. Mangasarian, E. W. Wild, and G. M. Fung, "Proximal knowledge-based classification," *Statistical Analysis and Data Mining*, vol. 1, no. 4, pp. 215–222, 2009.

[12] O. L. Mangasarian and E. W. Wild, "Nonlinear knowledge-based classification," *IEEE Transactions on Neural Networks*, vol. 19, no. 10, pp. 1826–1832, 2008.

[13] A. M. Bagirov, A. M. Rubinov, and J. Yearwood, "Using global optimization to improve classification for medical diagnosis and prognosis," *Topics in Health Information Management*, vol. 22, no. 1, pp. 65–74, 2001.

[14] Y. Liu, G. Wang, H. Chen, H. Dong, X. Zhu, and S. Wang, "An improved particle swarm optimization for feature selection," *Journal of Bionic Engineering*, vol. 8, no. 2, pp. 191–200, 2011.

[15] I. Iguyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[16] P. S. Bradley and O. L. Mangasarian, "Feature selection via concave minimization and support vector machines," in *Proceedings of the 15th International Conference on Machine Learning (ICML '98)*, Morgan Kaufmann, San Francisco, Calif, USA, 1998.

[17] A. M. Bagirov, B. Ferguson, S. Ivkovic, G. Saunders, and J. Yearwood, "New algorithms for multi-class cancer diagnosis using tumor gene expression signatures," *Bioinformatics*, vol. 19, no. 14, pp. 1800–1807, 2003.

[18] A. M. Bagirov, A. M. Rubinov, and J. Yearwood, "A heuristic algorithm for feature selection based on optimization techniques," in *Heuristic and Optimization for Knowledge Discovery*, R. A. Sarker, H. A. Abbas, and C. Newton, Eds., pp. 13–26, Idea Group Publishing, London, UK, 2002.

[19] C. Audet and J. E. Dennis Jr., "Mesh adaptive direct search algorithms for constrained optimization," *SIAM Journal on Optimization*, vol. 17, no. 1, pp. 188–217, 2007.

[20] A. M. Bagirov, "Modified global k-means algorithm for minimum sum-of-squares clustering problems," *Pattern Recognition*, vol. 41, no. 10, pp. 3192–3199, 2008.

[21] M. Fathian, B. Amiri, and A. Maroosi, "Application of honey-bee mating optimization algorithm on clustering," *Applied Mathematics and Computation*, vol. 190, no. 2, pp. 1502–1513, 2007.

[22] A. Likas, N. Vlassis, and J. J. Verbeek, "The global k-means clustering algorithm," *Pattern Recognition*, vol. 36, no. 2, pp. 451–461, 2003.

[23] M. Kudo and J. Sklansky, "Comparison of algorithms that select features for pattern classifiers," *Pattern Recognition*, vol. 33, no. 1, pp. 25–41, 2000.

[24] Z. Zhao, J. C. Meza, and M. V. Hove, "Using pattern search methods for surface structure determination of nanomaterials," *Journal of Physics: Condensed Matter*, vol. 18, pp. 8693–8706, 2006.

[25] C. L. Blake and C. J. Merz, "UCI repository of machine learning databases," 1998, http://archive.ics.uci.edu/ml/datasets.html.

[26] O. L. Mangasarian, "Misclassification minimization," *Journal of Global Optimization*, vol. 5, no. 4, pp. 309–323, 1994.

[27] K. P. Bennett and O. L. Mangasarian, "Robust linear programming discrimination of two linearly inseparable sets," *Optimization Methods and Software*, vol. 1, no. 1, pp. 23–34, 1992.

[28] C. Chen and O. L. Mangasarian, "Hybrid misclassification minimization," Mathematical Programming Technical Report no. 95–05, University of Wisconsin, 1995.

[29] P. S. Bradley and O. L. Mangasarian, "Feature selection via concave minimization and support vector machines," in *Proceedings of the 15th International Conference on Machine Learning (ICML '98)*, 1998.

[30] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "Improvements to Platt's SMO algorithm for SVM classifier design," *Neural Computation*, vol. 13, no. 3, pp. 637–649, 2001.

[31] J. C. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods: Support Vector Learning*, pp. 185–208, MIT Press, Cambridge, Mass, USA, 1999.

[32] I. de Falco, A. Della Cioppa, and E. Tarantino, "Facing classification problems with Particle Swarm Optimization," *Applied Soft Computing Journal*, vol. 7, no. 3, pp. 652–658, 2007.

[33] H. A. Firpi and E. Goodman, "Swarmed feature selection," in *Proceedings of the 33rd Applied Imagery Pattern Recognition Workshop*, pp. 112–118, IEEE, Washington, DC, USA, October 2005.

[34] Z. W. Geem, *Music-Inspired Harmony Search Algorithm: Theory and Applications*, Springer, New York, NY, USA, 1st edition, 2009.

[35] H. Banati and M. Bajaj, "Fire fly based feature selection approach," *International Journal of Computer Science Issues*, vol. 8, no. 4, pp. 473–480, 2011.

[36] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tool and Technique with Java Implementation*, Morgan Kaufmann, San Francisco, Calif, USA, 2000.

[37] M. H. Hassoun, *Fundamentals of Artificial Neural Networks*, The MIT Press, 1995.

[38] J. G. Cleary and L. E. Trigg, "K^*: an instance-based learner using an entropic distance measure," in *Proceedings of the 12th International Conference on Machine Learning*, pp. 108–114, 1995.

[39] P. Compton and R. Jansen, "Knowledge in context: a strategy for expert system maintenance," in *Proceedings of the 2nd Australian Joint Artificial Intelligence Conference (Ai '88)*, Lecture Notes in Computer Science, Adelaide, Australia, November 1988.

[40] G. Demiröz and A. Güvenir, "Classification by voting feature intervals," in *Machine Learning: ECML-97*, vol. 1224 of *Lecture Notes in Computer Science Volume*, pp. 85–92, 1997.

[41] R. Kohavi, "Scaling up the accuracy of naive-bayes classifiers: a decision tree hybrid," in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, AAAI Press, 1996.

[42] F. Jensen, *An Introduction to Bayesian Networks*, UCL Press/Springer, Berlin, Germany, 1996.

[43] N. Jankowski and V. Kadirkamanathan, "Statistical control of RBF-like networks for classification," in *Proceedings of the 7th International Conference on Artificial Neural Networks (ICANN '97)*, Lausanne, Switzerland, 1997.

[44] W.-H. Au and K. C. C. Chan, "Classification with degree of membership: a fuzzy approach," in *Proceedings of the 1st IEEE International Conference on Data Mining (ICDM '01)*, pp. 35–42, December 2001.

[45] L. Rutkowski and K. Cpalka, "Flexible neuro-fuzzy systems," *IEEE Transactions on Neural Networks*, vol. 14, no. 3, pp. 554–574, 2003.

[46] W. D. Leon, "Enhancing pattern classification with relational fuzzy neural networks and square BK-products," in *Computer Science*, pp. 71–74, Springer, New York, NY, USA, 2006.

[47] D. T. Pham, S. S. Dimov, and Z. Salem, "Technique for selecting examples in inductive learning," in *Proceedings of the European Symposium on Intelligent Techniques (ESIT '00)*, Aachen, Germany, 2000.

[48] W. P. Eklund and A. Hoang, "Comparative study of public-domain supervised machine-learning accuracy on the UCI database," in *Data Mining and Knowledge Discovery: Theory, Tools, and Technology*, V. D. Belur, Ed., vol. 3695 of *Proceedings of SPIE*, pp. 39–50, 1999.

[49] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Machine Learning*, vol. 29, no. 2-3, pp. 131–163, 1997.

[50] H. J. George and P. Langley, "Estimating continuous distributions in Bayesian classifiers," in *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence*, pp. 338–345, Morgan Kaufman, San Mateo, Calif, USA, 1995.

[51] N. Cheung, *Machine learning techniques for medical analysis [M.S. thesis]*, School of Information Technology and Electrical Engineering, University of Queensland, Queensland, Australia, 2001.

[52] Y.-J. Lee and O. L. Mangasarian, "SSVM: a smooth support vector machine for classification," *Computational Optimization and Applications*, vol. 20, no. 1, pp. 5–22, 2001.

[53] Y. J. Lee and O. L. Mangasarian, "RSVM: reduced support vector machines," in *Proceedings of the 1st SIAM International Conference on Data Mining*, Chicago, Ill, USA, 2001.

[54] T. van Gestel, J. A. K. Suykens, G. Lanckriet, A. Lambrechts, B. de Moor, and J. Vandewalle, "Bayesian framework for least-squares support vector machine classifiers, Gaussian processes, and Kernel fisher discriminant analysis," *Neural Computation*, vol. 14, no. 5, pp. 1115–1147, 2002.

[55] E. Çomak, K. Polat, S. Güneş, and A. Arslan, "A new medical decision making system: least square support vector machine (LSSVM) with fuzzy weighting pre-processing," *Expert Systems with Applications*, vol. 32, no. 2, pp. 409–414, 2007.

[56] J. Tian, M. Li, and F. Chen, "Dual-population based coevolutionary algorithm for designing RBFNN with feature selection," *Expert Systems with Applications*, vol. 37, no. 10, pp. 6904–6918, 2010.

[57] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proceedings of the 13th International Conference on Machine Learning*, pp. 148–156, San Francisco, Calif, USA, 1996.

[58] E. Frank and I. H. Witten, "Generating accurate rule sets without global optimization," in *Proceedings of the 15th International Conference on Machine Learning*, J. W. Shavlik, Ed., pp. 144–151, Morgan Kaufmann, San Francisco, Calif, USA, 1998.

[59] C. Jesus and L. M. Ramon, "The indifferent naive Bayes classifier," in *Proceedings of the 16th International FLAIRS Conference*, pp. 341–345, 2003.

[60] K. Polat, S. Şahan, H. Kodaz, and S. Güneş, "Breast cancer and liver disorders classification using artificial immune recognition system (AIRS) with performance evaluation by fuzzy resource allocation mechanism," *Expert Systems with Applications*, vol. 32, no. 1, pp. 172–183, 2007.

[61] N. García-Pedrajas and D. Ortiz-Boyer, "Improving multiclass pattern recognition by the combination of two strategies," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 6, pp. 1001–1006, 2006.

[62] D. Michie and D. J. Spiegelhalter, *Machine Learning, Neural and Statistical Classification*, Series in Artificial Intelligence, Ellis Horwood, London, UK, 1994.