Information Sciences 626 (2023) 586–606

Contents lists available at ScienceDirect

Information Sciences

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

Feature Selection Using Diversity-Based Multi-objective Binary Differential Evolution

Peng Wang [a], Bing Xue [a], Jing Liang [b,c,*], Mengjie Zhang [a]

[a] School of Engineering and Computer Science, Victoria University of Wellington, Wellington 6012, New Zealand
[b] School of Electrical Engineering and Automation, Henan Institute of Technology, Xinxiang 453000, China
[c] School of Electrical and Information Engineering, Zhengzhou University, Zhengzhou 450001, China

ARTICLE INFO

Keywords:
multi-objective optimization
differential evolution
feature selection
population diversity

ABSTRACT

By identifying relevant features from the original data, feature selection methods can maintain or improve the classification accuracy and reduce the dimensionality. Recently, many multi-objective evolutionary methods have been proposed for feature selection. However, effectively handling the trade-offs between convergence and diversity of the non-dominated solutions remains a major challenge, especially for high-dimensional datasets. To cover this issue, this work studies a diversity-based multi-objective differential evolution approach to feature selection. During the environmental selection process, each of the solutions in the candidate pool will have a diversity score, and solutions with large diversity score values will be preferred so as to improve the population diversity. To reduce the search space, irrelevant and weakly relevant features are detected and removed in the proposed method. A new binary mutation operator using the neighborhood information of individuals is also proposed, aiming to produce better feature subsets. Experimental results on 14 datasets with varying difficulties show that the proposed feature selection method can obtain significantly better feature selection performance than current popular multi-objective feature selection methods.

1. Introduction

To prevent the loss of important information, data collected from real-world scenarios usually contains a large number of features. Data with high dimensionality needs more computational and storage resources. Furthermore, noisy, redundant, and/or irrelevant features in the data lead to unsatisfactory classification performance. Feature selection [1] which can effectively select relevant features from the original feature set has been an important research focus. Generally, a feature selection task has two main objectives: decreasing the number of features selected and increasing the classification accuracy (decreasing the classification error rate), which are potentially conflicting with each other [2].

Over the past decades, evolutionary computation (EC) methods [3] including evolutionary multi-objective optimization (EMO) technique [2] have been popular in designing feature selection approaches. The population-based search mechanism is particularly suitable for handling feature selection tasks since it can find a set of non-dominated solutions (feature subsets) with the trade-off between different conflicting objectives in a single run. Compared with single-objective feature selection algorithms, EMO-based

* Corresponding author.
E-mail addresses: wangpeng@ecs.vuw.ac.nz (P. Wang), bing.xue@ecs.vuw.ac.nz (B. Xue), liangjing@zzu.edu.cn (J. Liang), mengjie.zhang@ecs.vuw.ac.nz (M. Zhang).

https://doi.org/10.1016/j.ins.2022.12.117
Received 17 September 2022; Accepted 30 December 2022
Available online 3 January 2023

feature selection methods can avoid the setting of sensitive parameters [4]. Existing EC-based and/or EMO-based feature selection methods mainly include genetic algorithms (GAs) [5–7], differential evolution (DE) [8–10], particle swarm optimization (PSO) [11,12], and genetic programming (GP) [13] to name a few. Given the fact that DE has the advantages of simplicity and efficiency [14], many DE-based feature selection methods have been developed. To this end, the EMO-based DE techniques have been suggested to solve feature selection tasks [14,15]. Although these methods have shown their competitiveness in getting good feature selection performance, most of them suffer from high computation costs or getting trapped into local optima. The reason is that there exists a huge search space and complex feature interactions, especially for high-dimensional datasets [16].

Generating high-quality offspring is one of the main factors affecting the final feature selection performance of EC-based feature selection methods [17]. There are already many studies studying the update strategies in EMO-based feature selection methods, such as binary DE-based mutation operator [14], steering-matrix-based evolutionary operators [4], and granularity-based mutation and crossover operators [18]. However, these update strategies ignore the information of neighbors of individuals, and therefore some good feature subsets may be lost during evolution. To overcome this limitation, some methods such as [19,9,15] employ the neighbors' information of individuals in the population to generate new solutions. However, these methods suffer from high computation costs especially on high-dimensional datasets because the calculation of crowding distance in the search or solution space is frequently required when selecting individuals to form a new population, i.e., environmental selection. Another key point is to consider the population diversity in the solution space in environmental selection. Maintaining a good population diversity can better explore the search space, thereby reducing the possibility of generating duplicated feature subsets and helping an algorithm jump out of local optima.

Furthermore, the removal of irrelevant features and weakly relevant features in a dataset is also very important for a feature selection task. By removing such types of features, the search space of a feature selection task can be decreased, and more computation resources can be saved to explore other promising regions. Irrelevant features such as features with a variance of zero are easy to detect. For the weakly relevant features, information theory is often used to quantify the relative importance of each feature in data. Many studies such as [20,12,21] treat features with lower correlations to the class labels than a predefined threshold as the weakly relevant features. However, other essential information such as the intra-class compactness and inter-class separation induced by features should also be considered when identifying weakly relevant features. Some features force very large distances between instances in the same class and/or force very small distances between instances in different classes. Most likely, these features may affect the generalization performance on unseen/test data if they are selected, especially for distance-based or similarity-based learners.

To overcome these limitations, a diversity-based multi-objective binary DE algorithm (termed DMBDE) is proposed in this work. The major contributions are summarized below:

1) A new method is developed that can effectively detect and remove weakly relevant features. During the detection stages, both the relevance between features and the class labels and the contribution of each feature to the intra-class compactness and inter-class separation are considered. The results indicate that the developed removal mechanism can improve the feature selection performance of the proposed method.
2) Under the EMO framework, a new binary mutation operator in DE is proposed. The neighbors' information of solutions is considered when generating mutant vectors. The neighborhood size of an individual is automatically determined. The results show that the proposed mutation operator can generate feature subsets with better fitness values.
3) An environmental selection mechanism by calculating the diversity scores of individuals in the solution space is proposed. The solutions with large diversity scores are preferred during the environmental selection process. Crowding distance in the objective space is regarded as the secondary criterion, activated when the diversity value of individuals is the same. The results show that the proposed environmental selection mechanism can improve the final classification performance.

The remainder of this paper is structured below. Section 2 introduces the related work. The proposed DMBDE algorithm is introduced in Section 3, and the experimental settings are given in Section 4. Section 5 presents the results. Lastly, this paper is concluded in Section 6.

2. Related Work

2.1. Multi-objective Optimization for Feature Selection

A multi-objective optimization problem needs to simultaneously optimize T objectives, which is defined below:

$$\begin{array}{ll} \min & \vec{f}(\vec{x}) = (f_1(\vec{x}), f_2(\vec{x}), \ldots, f_T(\vec{x})) \\ \text{s. t.} & g_i(\vec{x}) \leqslant 0 \quad i = 1, 2, \ldots, g \end{array} \tag{1}$$

where $\vec{x} \in \Omega$ represents a solution to a problem or task with D-dimensional solution space Ω. Meanwhile, $\vec{f}(\vec{x})$ is a vector including T objective functions, and $g_i(\vec{x})$ is the constraint functions of the problem. In feature selection tasks, simultaneously minimizing the number of the selected features and minimizing the classification error rate of employing the selected features are the two main objectives. In an EC method, a solution to a feature selection task can have the following form:

$$\overrightarrow{x} = (x_1, \ldots, x_D) \text{ and } x_j = 0 \text{ or } 1,\ j \in [1, D] \tag{2}$$

where D means the dimensionality of a dataset, and $x_j = 1$ means the j-th feature in the dataset is selected while $x_j = 0$ means not. After calculating the number of 1s, i.e., the number of the selected features (termed k), FR equals k/D representing the ratio of the number of the selected features (k) over the total number (D) of the original features in a dataset, and ER means the classification error rate. Along this way, the two objective values are within the range $(0, 1)$. More specifically, Eq. (3) shows the objective functions:

$$\min \begin{cases} f_1 = \mathrm{FR} = k/D \\ f_2 = \mathrm{ER} = \dfrac{FP + FN}{TP + TN + FP + FN} \end{cases} \tag{3}$$

where FP, FN, TP, and TN are the false positives, false negatives, true positives, and true negatives, respectively.

2.2. Filter Measures in Feature Selection

2.2.1. Mutual Information (MI)

MI [22] is a typical technique to measure the relevance between paired variables, e.g., X and Y, as shown in Eq. (4):

$$\mathrm{MI}(X; Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \tag{4}$$

where $p(x), p(y)$, and $p(x, y)$ respectively mean the probability distribution of x, y, and their joint distribution. If X and Y are two features, $\mathrm{MI}(X; Y)$ measures how much information is shared between two features. If X is a feature and Y is the class labels, $\mathrm{MI}(X; Y)$ measures the degree of dependence of feature X to the class labels Y [23].

2.2.2. Maximal Information Coefficient (MIC)

MIC [24] is a recently proposed statistical measure, using bins to apply MI on continuous variables. By dividing variable values into different grids and normalizing MI, MIC is observed to be able to measure the relationships between variables more accurately [24,25]. The calculation of MIC is introduced briefly below. The original space G (the values of X and Y) are divided into multiple p-by-q grids. The characteristic matrix $\mathrm{M(G)}_{p,q}$ means the highest normalized MI of G in the $p \times q$ partitions, as shown in Eq. (5):

$$\mathrm{M(G)}_{p,q} = \frac{\max(\mathrm{MI})}{\log \min\{p, q\}} \tag{5}$$

where $\max(\mathrm{MI})$ denotes the maximal MI of G within all the $p \times q$ partitions. Next, MIC is defined as:

$$\mathrm{MIC} = \max_{0 < p \times q < B(n)} \{\mathrm{M(G)}_{p,q}\} \quad B(n) = n^{0.6} \tag{6}$$

where n means the number of instances, and $B(n)$ is the constraint of the grid size ($p \times q$). More details and sensitivity analysis of parameters in MIC can be seen in [24].

The value of MIC tends to be 1 for two highly correlated variables and tends to be 0 for two statistically independent variables. Therefore, the greater the value, the higher the dependence between paired variables.

2.2.3. Fisher Score

Fisher score [26] can rank features in a dataset by maximizing the variance between classes and minimizing the variance within classes. Let μ^j and σ^j be the mean and the standard deviation of the j-th feature (F_j) from the whole data. Then, the mean and standard deviation of F_j from the k-th class are termed μ_k^j and σ_k^j, respectively. The Fisher score of F_j is computed below:

$$\mathrm{Fisher}(F_j) = \frac{\sum_{k=1}^{c} n_k (\mu_k^j - \mu^j)^2}{(\sigma^j)^2} \tag{7}$$

where $(\sigma^j)^2 = \sum_{k=1}^{c} n_k (\sigma_k^j)^2$, and n_k is the number of instances in the k-th class. Generally, features with higher Fisher scores are more discriminant than features with lower scores.

2.2.4. Further Example for Calculating MIC

MIC is a more complex concept than MI and Fisher Score. Intuitively, MIC is based on the idea that if there is a relationship between two variables, different grids/bins can be drawn on the scatter plot of the two variables, partitioning the data to encapsulate the relationship [24]. An example is given in Fig. 1 to show its calculation process.

For a set of two-variable data (termed G in Fig. 1), the first step is to explore all grids up to a maximal grid resolution $B(n)$. After calculating each pair of integers (p, q), the MIC algorithm will find the p-by-q grid with the highest induced mutual information. In Fig. 1A, the situation with $p = 2$ and $q = 2$ is shown, and the second grid division in Fig. 1A is supposed to have the largest MI value achievable by any 2-by-2 grid. Next, the MI values will be normalized followed by Eq. (5). The normalized values will form a characteristic matrix storing the best grid at that resolution and its normalized score, as shown in Fig. 1B. Finally, MIC will be denoted as the maximal value in the matrix, as shown in Eq. (6).

2.3. Feature Selection Methods

Feature selection approaches can be grouped into three main categories: wrapper, filter, and embedded methods, according to the involvement of a learning algorithm in the selection process [3]. In wrapper methods, learning algorithms are employed to evaluate the classification performance of a feature subset. Filter methods are independent of any learning algorithm, often using statistical measures, for instance MI, to determine the goodness of a feature subset. In embedded methods, feature selection is embedded in the learning process of a classifier. Additionally, hybrid methods such as filter-wrapper hybrid recently are investigated to use correlation knowledge from some statistical measures to help the search [21].

2.3.1. Typical Feature Selection Methods

Recently, many feature selection methods including EC-based feature selection methods have been proposed. These methods try to enhance the performance on solving feature selection tasks by considering feature redundancy, feature relevance, and/or designing novel search operators.

Some filter methods, for example, ReliefF [27] and the minimum redundancy maximum relevance (mRMR) principle [28], combined several highly ranked features which may produce redundancy and fall into a local optimum. Although some improvements have been proposed such as [29,30], they may still require a user to provide the number of the selected features. EC methods have recently gained much attention since they do not need domain knowledge [3]. Salesi et al. [31] designed a two-stage filter method with GA for feature selection. The first stage uses Fisher score to remove features. The second stage employs GA to select the final subset of features by optimizing the mRMR principle. However, the used redundancy measure considers only two-way interaction. Ma et al. [13] applied GP to simultaneous feature selection and feature construction. A MI-based filter measure is designed to evaluate the constructed high-level features. However, the study focused on datasets mainly including tens to hundred of features. Chen et al. [16] employed ReliefF to obtain the weight of each feature in a dataset, and the generation of part of new solutions in PSO is guided by the weights' information. However, irrelevant features in a dataset can still have the chance to be selected. To overcome this limitation, Song et al. [21] applied PSO to select informative features, and the irrelevant and weakly relevant features are removed firstly. However, the feature subset obtained by [21] may not adequately represent the original data since only one feature is allowed to select from each group after grouping features.

Hancer et al. [32] proposed a cost-sensitive filter method with DE for feature selection. The proposed method supposed that features in a dataset have different feature costs. However, the proposed method does not explore its performance on high-dimensional datasets. To solve high-dimensional feature selection tasks, Tarkhaneh et al. [10] proposed a wrapper-based DE method with modified mutation and crossover mechanisms. During training, the developed mechanisms employ the best-so-far individual to produce new solutions. The results show that the proposed mechanisms can help the method have superiority against the comparison algorithms. However, the performance of the proposed method might be sensitive to two parameters in the fitness function used to balance the classification error rate and the subset size.

2.3.2. EMO Techniques for Feature Selection

By simultaneously minimizing the two objectives: the classification error rate and the number of the selected features, EMO techniques can output a set of feature subsets to a user. Followed by the main process of NSGA-II [33], Li et al. [7] applied so-called direct multisearch to feature selection. The key idea is to produce new solutions around the current non-dominated solutions. However, the study focused mainly on small-scale feature selection tasks. To improve the search efficiency for high-dimensional datasets, Cheng et al. [4] employed a so-called steering matrix for feature selection. The proposed SM-MOEA method can significantly reduce

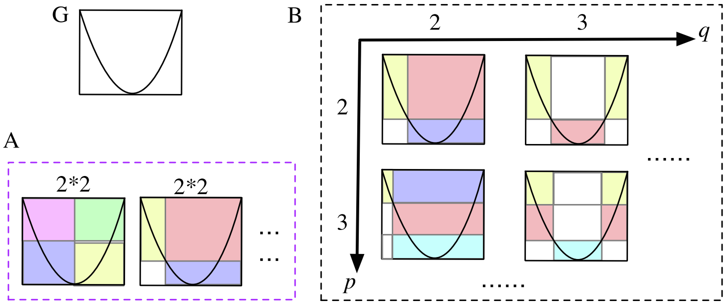

Fig. 1. An example to calculate MIC between paired variables.

the subset size and achieve good accuracy results on 12 high-dimensional datasets. Li et al. [34] designed a binary individual search strategy-based EMO feature selection method, termed BIBE. BIBE removes the irrelevant and redundant features with an improved Fisher score. A binary crossover operator and a binary mutation operator are proposed to generate new individuals. However, BIBE ignores the negative impact of duplicated solutions on the final feature selection performance. To generate as few duplicated feature subsets as possible, Xu et al. [17] proposed a duplicated analysis-based feature selection method, termed DAEA. DAEA uses Manhattan distance in the solution space to measure the similarity degree of solutions. The results showed that the proposed reproduction strategy contributes the most to the good performance of DAEA. Zhang et al. [14] proposed a binary mutation operator. In the proposed mutation operator, one of the three individuals randomly chosen from the whole population is considered the base vector, while the remaining two individuals are used to obtain mutation probabilities. However, such a mutation operator may slow down the convergence due to without considering the distance or fitness among the chosen individuals. In addition, applying niching techniques to EMO methods aiming to find multiple optimal feature subsets is another research focus. Wang et al. [9] proposed a multi-objective DE with K-means clustering technique (MOCDE) for feature selection. Although MOCDE obtained promising results, some redundant features may still be selected. To further reduce the redundant rate of the selected features, a niching-based multi-objective DE (NMDE) is proposed in [15]. NMDE achieves significantly better HV and IGD performance than the compared methods. However, both MOCDE and NMDE have high computational costs due to the complex environmental selection process. The methods SM-MOEA, DAEA, and NMDE can show generally better feature selection performance than other EMO-based feature selection methods. Therefore, the three methods will be adopted to compare with our proposed method.

3. Proposed Method

The general scheme of the proposed multi-objective binary differential evolution algorithm (DMBDE) is described first. Then, some essential components of DMBDE including the removal operator, the generations of offspring, and the diversity-based environmental selection mechanism are discussed.

3.1. General Scheme

Alg. 1 describes the main process of DMBDE. Firstly, irrelevant and weakly relevant features are detected and removed from the original feature set. The details are shown in Section 3.2. The next step is the initialization of DMBDE. The initialization mechanism of DMBDE is the same as that in DAEA [17]. Specifically, DMBDE will randomly select features without any restriction when the subset size from an individual is less than three times the population size (termed 3N). Otherwise, the number of the selected features in an individual in DMBDE is required to be no more than 3N. Since it is difficult to cover the entire solution space, especially on high-dimensional datasets, reducing the subset size to 3N is a compromised way for generating the initial population.

Algorithm 1: DMBDE

```
Input: Population size $N$ and termination condition $t_{\max}$
Output: Population $P$
begin
    Calculate the Fisher score and MIC value of each feature,
    Get the importance score of each feature via Eq. (8),
    Save the features whose Score values are larger than $\delta$,
    Get the initial population $P$,
    Set the objective function evaluation number $t = N$,
    while $t < t_{\max}$ do
        Get new individuals by Alg. 2,
        Put all new individuals into offspring set $O$,
        $t = t + N$,
        Perform environmental selection,
        Select individuals and form a new population $P$,
    end
    Output feature subsets in the first front.
end
```

After initialization, the generation of new individuals is shown in Section 3.3. During the environmental selection process, all duplicated feature subsets in the solution space are removed. The non-dominated sorting and the diversity maintenance are shown in

Section 3.4. When the maximal number of evaluations is met, DMBDE will output the solutions at the first front.

3.2. Removal of Irrelevant and Weakly Relevant Features

For a feature set $\mathscr{F}$ including D original features, $\mathscr{F} = \{F_1,\ldots,F_j,\ldots,F_D\}$, the importance score of each feature F_j in $\mathscr{F}$ is determined as follows:

$$\text{Score}(F_j) = 0.5 * \text{MIC}(F_j) + 0.5 * \text{Fisher}(F_j) \tag{8}$$

where $\text{MIC}(F_j)$ measures the dependency of feature F_j to the class labels, and $\text{Fisher}(F_j)$ calculates the score of feature F_j by considering the intra-class and the inter-class distances. After calculating MIC values and Fisher scores for all features, both $\text{MIC}(F_j)$ and $\text{Fisher}(F_j)$ are normalized by using the min-max normalization technique, respectively. There is no preference for the two parts, so Eq. (8) takes 0.5 as the coefficient.

After obtaining the Score values of all features, the next step is to delete irrelevant and weak relevant features from the original feature set in order to reduce the size of the feature space. Note that the more relevant the feature, the greater the Score value. A small constant δ can be set to remove the features with low Score values. In other words, only the features with Score values greater than δ are kept. The value of δ is determined by Eq. (9):

$$\delta = a * \text{Score}_{\max} \tag{9}$$

where a is a small coefficient and $\text{Score}_{\max}$ means the maximal value of Score value in a dataset. Section 4 will conduct a sensitivity analysis to determine the appropriate value for a.

3.3. Generation of Offspring

As shown in Alg. 2, the generation process of offspring in DMBDE has main three steps: the introduction of niching behavior (Lines 3-4), the binary mutation (Lines 6-11), and crossover (Line 12).

The generation of new individuals in DMBDE is based on binary encoding which is more straightforward than real-valued encoding for feature selection [35,36] since feature selection tasks are inherently combinatorial optimization problems (i.e., select or not select a feature). In addition, continuous DE methods using a real-valued encoding such as [37,38] are more likely to generate duplicated feature subsets since one extra strategy is needed to convert solutions from the search space to the solution space. In [14], a binary DE feature selection method was proposed, but no neighbor information is considered when generating offspring. To further improve the feature selection performance, a newly developed binary mutation operator with niching techniques is proposed. Niching techniques have been used to facilitate the exchange of evolutionary information stably and efficiently [17].

Specifically, DMBDE normalizes the objective values of solutions in each of the objective directions separately to the same scale in order to choose neighboring solutions fairly. Moreover, the niche size of each solution is automatically determined by using the three-sigma rule [39]. Specifically, three nearest neighbors of each individual (e.g., $\vec{x}_i$) will be its initial niche $Nei(i)$ (Line 3 of Alg. 2). Then, the average value (μ_i) and the variance value (σ_i) of the distances of solutions in $Nei(i)$ to $\vec{x}_i$ are calculated. Next, other individuals in the population fitted within the range $\mu_i \pm 3\sigma_i$ will be also included in the niche $Nei(i)$. As a result of the property of the univariate Gaussian distribution, 99.73% of the samples approximately lie within the range [40]. Consequently, when other individuals are not within the range $\mu_i \pm 3\sigma_i$, they cannot be considered to fit the niche.

The mutation operator randomly chooses parents with a proportion of 80% or 20% from an individual's niche or the whole population, which is a regular setting [17]. Then, a mutant vector will be generated using Eqs. (10) and (11):

$$v_{i,j} = \begin{cases} x_{\text{lbest},j} & \text{if } C_{i,j} < \text{rand} \\ 1 - x_{\text{lbest},j} & \text{otherwise} \end{cases} \tag{10}$$

$$C_{i,j} = \begin{cases} \sigma & \text{if } \vec{x}_{\text{lbest}} \prec \vec{x}_i \\ \min(1, F(x_{r_1,j} \oplus x_{r_2,j}) + \sigma) & \text{otherwise} \end{cases} \tag{11}$$

where $\vec{x}_{\text{lbest}}$ is the non-dominated one among the three randomly chosen vectors, and $\vec{x}_{r_1}$ and $\vec{x}_{r_2}$ are the remaining two vectors. If there are more than one non-dominated vectors, a randomly chosen vector from the multiple non-dominated vectors will be considered $\vec{x}_{\text{lbest}}$. Meanwhile, $\vec{x}_{\text{lbest}} \prec \vec{x}_i$ indicates that $\vec{x}_{\text{lbest}}$ dominates $\vec{x}_i$. The sign "$\oplus$" means XOR function, and $C_{i,j}$ is a probability value generated by running XOR on $x_{r_1,j}$ and $x_{r_2,j}$. The parameter σ is a small turbulence coefficient, and $F \in (0, 1]$ is a scale parameter which controls the learning rate of an individual from $\vec{x}_{\text{lbest}}$.

Algorithm 2: Mutation and Crossover

(continued on next page)

(*continued*)

```
Input: Population $P$
Output: Offspring $O$
begin
    Set $O = P$, and get $N = |P|$,
    Produce a neighborhood set $Nei$. $Nei(i)$ includes three nearest solutions to $P(i)$ based on
      the Euclidean distances among the normalized objective values,
    Update $Nei$ using three-sigma rule of thumb,
    for $i = 1, \ldots, N$ do
        if $\text{rand} < 0.8$ then
            Randomly select three individuals from $Nei(i)$,
        else
            Randomly select three individuals from $P$,
        end
        Generate a mutant vector via Eqs. (10)-(11),
        Perform crossover operator via Eq. (12),
        Set the new offspring as $O(i)$
    end
end
```

In Eq. (11), when $\vec{x}_{\text{lbest}} \prec \vec{x}_i$, i.e., $\vec{x}_i$ is dominated by $\vec{x}_{\text{lbest}}$, the mutant vector of $\vec{x}_i$ tends to learn more from $\vec{x}_{\text{lbest}}$. This may increase the probability that $\vec{x}_i$ becomes a non-dominated solution. When $\vec{x}_{\text{lbest}}$ is dominated by $\vec{x}_i$, the probability difference $F * (x_{r_1,j} \oplus x_{r_2,j}) + \sigma$ is calculated. Adding the difference between $\vec{x}_{r_1}$ and $\vec{x}_{r_2}$ can help maintain the diversity of the generated mutation vectors. To make sure the mutation probability is greater than 0, a small coefficient σ is used. This is to prevent a new solution from falling into a local optimum. The results in [14] indicate that $\sigma \in [0.001, 0.01]$ is appropriate. Without losing generality, σ is set to 0.005 in DMBDE.

The crossover operator in DMBDE is the same as in standard DE, as shown in Eq. (12):

$$u_{i,j} = \begin{cases} v_{i,j}, & \text{if } (\text{rand}(0,1) \leqslant CR) \text{ or } (j = j_{\text{rand}}) \\ x_{i,j}, & \qquad\qquad\qquad \text{otherwise} \end{cases} \tag{12}$$

where j_{rand} is a random integer, and $CR \in [0, 1]$ means crossover rate which controls the information learned from a mutant vector.

3.4. Diversity-based Environmental Selection

Fig. 2 shows the process of environmental selection. The main idea is to select individuals with higher diversity/dissimilar scores

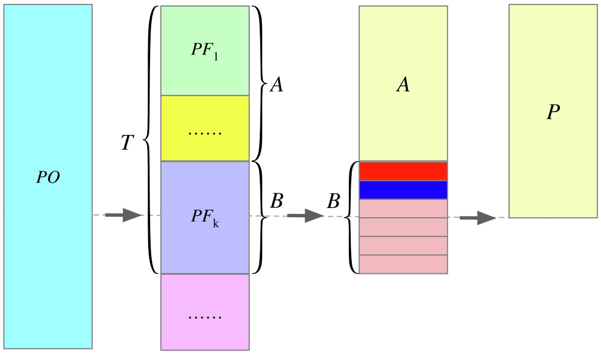

Fig. 2. Environmental selection process. After performing the non-dominated sorting scheme for PO, A includes the solutions in the first $k-1$ fronts, and B includes the solutions in the k-th front. Meanwhile, $T = A \cup B$. In B, each square represents a single solution, and the squares with the same color mean that these solutions have the same diversity score. In this example, the last four solutions in B have the same diversity score.

(d_s) to form a new population. It mainly includes two steps. All solutions from the parental set P and the offspring set O are combined, termed PO. After removing duplicated feature subsets in PO, the first step is to rank the solutions based on the non-dominated sorting scheme from [33]. The second step is to select solutions from the sorted fronts of PO as a new population P, starting from the first front. If adding the k-th front, the population size becomes larger than N, and the diversity scores of solutions in the k-th front will be calculated.

The second step is essential, and the detailed introductions will be given below. In Fig. 2, A is a set including all solutions in the first $k-1$ fronts. Candidate pool B includes all solutions in the k-th front ($B = PF_k$). Therefore, $T = A \cup B$ will represent all solutions in the first k fronts. Suppose that an individual $\vec{x}_i$ is in B, the d_s value of $\vec{x}_i$ is determined by Eqs. (13) and (14):

$$d_s(\vec{x}_i) = 1 - \cos(\vec{x}_i, \vec{x}_m) \tag{13}$$

$$\cos(\vec{x}_i, \vec{x}_m) = \frac{\sum_{j=1}^{D} x_{i,j} \times x_{m,j}}{\sqrt{\sum_{j=1}^{D} (x_{i,j})^2}\sqrt{\sum_{j=1}^{D} (x_{m,j})^2}} \tag{14}$$

where $\vec{x}_m$ is the individual in T with the largest cosine similarity score with $\vec{x}_i$ ($i \neq m \in T$), and j means the j-th dimension of one individual, $j \in [1, D]$. The cosine similarity is defined as the cosine of the angle between vectors. Due to the binary encoding of the solutions (in positive space), the outcome is neatly bounded in $[0, 1]$.

In Eq. (13), one solution with a high cosine value will have a low d_s value since having a high cosine value indicates that the solution selects many common features for other solutions. Therefore, Eq. (13) can be used to measure the diversity/dissimilar degree of solutions. The solutions in B with larger d_s values are preferred. This is to improve the population diversity and reduce the probability to produce duplicated feature subsets.

After getting the d_s value of each solution in B, solutions will be ranked in descending order. Next, the top q solutions will be selected from $B, q = N - |A|$. However, multiple different solutions may have the same d_s value as x_q. In Fig. 2, this situation is represented by the same color from solutions in B. For example, the last four solutions in B have the same d_s value. Under this situation, the crowding distance of these solutions will be calculated. The solution with the largest crowding distance value will be considered x_q. Finally, the selected q solutions with A form a new population P.

3.5. Further Discussions

Suppose a dataset includes D features and there is a population with N individuals to solve a problem with M objectives and d dimensions of decision variables, the overall complexity of the proposed DMBDE method is analyzed below.

DMBDE mainly contains six parts: removal operator, initialization, mutation, crossover, evaluation, and diversity-based environmental selection schemes. The complexity of DMBDE mainly depends on the removal operator, the mutation operator, and the diversity-based environmental selection scheme. The removal operator includes the calculations of MIC values and the Fisher scores of all features in a dataset. Therefore, its computational complexity is $O(2D)$. The niching-based mutation operator employs the distances between individuals in the objective space. Therefore, it executes $O(M * N)$ basic operations in each generation. The environmental selection strategy considers the non-dominated sorting, diversity scores of solutions in the candidate pool B, and crowding distance in the objective space. The complexity of the fast non-dominated sorting is $O(M * N^2)$ [33]. The complexity of the calculation of the diversity scores in the solution space is related to the non-dominating sorting technique. When all population members are in one front, at most N solutions are involved. Under this situation, the computational complexity of the calculation of the diversity score is $O(d * N)$. The complexity of the calculation of the crowding distance in the objective space is $O(M * N * \log N)$ [33].

Given the fact that $M * N * \log N < M * N^2$, the overall complexity of DMBDE is approximately $O(2D + M * N^2 + d * N)$. Since some features are deleted by the removal operator, d is lower than D, and therefore the upper limit of the complexity of DMBDE is $O(2D + 2N^2 + D * N)$.

4. Experiments

4.1. Datasets

In the experiments, 14 datasets including data from different fields, e.g., physics (Sonar), chemistry (Musk1), medicine/health (Arrhythmia, WBCD), and handwritten recognition (Multiple) are tested. The 14 datasets have different numbers of features (30 to 15,009), classes (2 to 26), and instances (72 to 2,600). In addition, the six datasets, i.e., the SRBCT, Leukemia, Prostate, 11Tumor, LungCancer, and 14Tumor datasets, are from the biomedical domain, including thousands of features but tens of instances (SRBCT and Leukemia) and up to 15,009 features and a few hundreds of instances (Prostate, 11Tumor, LungCancer, and 14Tumor). Those 6 gene expression datasets pose a huge challenge to classification and feature learning algorithms, due to the high dimensionality but a small sample size. In addition, these datasets cover a variety of data types, e.g., real-value features, integer-value features, or a mixture of both types. The datasets are selected with the expectation that they can be well representatives of real-world problems. Table 1 lists the basic information of the 14 datasets. The first eight datasets in Table 1 are from the UCI machine learning repository [41], and the remaining six high-dimensional gene expression datasets are originally downloaded from http://www.gemssystem.org. All the used 14 datasets can be found at https://github.com/penfwang/Inf_Sci_MODE. Each dataset is randomly divided into a training set and a test set with ratios of 70% and 30%, respectively.

During training, 5-fold cross-validation is performed using a simple learning algorithm, i.e., KNN, to get the classification error rate for the selected features on the training set. Another reason to use KNN is that KNN does not have any assumption about the probability distribution of a dataset. Moreover, feature subsets produced by wrapping KNN can be generalized to other classification algorithms such as Decision Trees and Naïve Bayes [42].

4.2. Benchmark Techniques

To validate the effectiveness of the proposed algorithm, DMBDE is compared with seven feature selection methods. They are SPEA2 [43], MOEA/D-DE [44], GF-NSGAII [45], SparseEA [46], DAEA [17], SM-MOEA [4], and NMDE [15]. Among them, SPEA2 and MOEA/D-DE are two commonly used EMO algorithms. The methods GF-NSGAII, SparseEA, DAEA, SM-MOEA, and NMDE are five popular EMO-based feature selection methods.

4.3. Performance Indicators

For the eight EMO methods, the hypervolume (HV) indicator is used to show their comprehensive performance. HV is defined as the size of the space covered by the non-dominated solutions, as shown in Eq. (15):

$$\mathrm{HV}(A, r_p) = \mathscr{L}(\cup_{y\in A}\{x|y \prec x \prec r_p\}) \tag{15}$$

where A means the set of the non-dominated solutions and r_p represents a reference point. Meanwhile, $\mathscr{L}(\cdot)$ means the Lebesgue measure of a set [47], and $y \prec x$ denotes that x dominates y. The reference point r_p of HV is usually set to $(1,1)$ for a minimization optimization problem.

In addition, considering the possible imbalances in the tested datasets, the F1 score is used as another performance indicator. F1 score is the harmonic mean of precision and recall, as shown in Eq. (16):

$$\mathrm{F1} = \frac{2 * TP}{2 * TP + FP + FN} \tag{16}$$

where the typos FP, FN, and TP have the same meaning as that in Eq. (3).

Furthermore, for the feature subset with the lowest training classification error rate in each method, its average classification accuracy (termed Ac) on the test set and the average subsets size (termed d^*) are also reported. The Wilcoxon test with a significance level of 0.05 is used to judge whether the performance of DMBDE is significantly different from that of any other competing algorithms. For relative performance rankings, the Friedman test is also adopted.

4.4. Parameter Settings

Each feature selection algorithm will perform 30 times on each training set, independently. Table 2 lists the specific parameters of the algorithms. The maximal number of fitness evaluations is set to 100 (generations) times N (population size) for all the tested datasets. Meanwhile, N is set to the number of original features in a dataset but limited to 300 as suggested in [45]. Additionally, K is set to 5 in KNN to balance the accuracy and efficiency in a feature selection task [9,48].

For the small constant a in DMBDE, the average training HV performance of a at 0,0.05,0.1,0.15, and 0.2 are reported in Table 3. In Table 3, DMBDE with $a = 0.1$ achieves the highest HV results on 7 out of the 14 datasets among the five methods. Although a slightly

Table 1
The information of datasets

Index	Dataset	# Features	# Classes	# Instances	Feature Type	Area
1	WBCD	30	2	569	Real	Medicine
2	Sonar	60	2	208	Real	Physics
3	Movement	90	15	360	Real	Medicine
4	Hillvally	100	2	1,212	Real	Graph
5	Musk1	166	2	2,031	Integer	Physics
6	Multiple (pix)	240	10	2,000	Mixed	Digit Recognition
7	Arrhythmia	279	16	452	Mixed	Medicine
8	Madelon	500	2	2,600	Real	Artificial Dataset
9	SRBCT	2,308	4	83	Integer	Medicine
10	Leukemia	5,147	2	72	Integer	Medicine
11	Prostate	10,509	2	102	Mixed	Medicine
12	11Tumor	12,533	11	174	Mixed	Medicine
13	LungCancer	12,600	5	203	Real	Medicine
14	14Tumor	15,009	26	308	Mixed	Medicine

Table 2
Parameter Settings

Algorithms	Parameter Values
SPEA2 [43]	The mutation rate $= 1/D$, crossover probability $= 1$, threshold $\theta = 0.6$
MOEA/D-DE [44]	The mutation rate $= 1/D$, crossover rate $CR = 0.5$, selection probability of neighborhoods $= 0.9$, scale factor $F = 0.5$, threshold $\theta = 0.6$
GF-NSGAII [45]	The mutation rate $= 1/D$, crossover probability $= 1$, number of grids $div = 15$, threshold $\theta = 0.6$
SparseEA [46]	The mutation rate $= 1/D$, crossover probability $= 1$
DAEA [17]	The mutation rate $= 1/D$, the crossover rate $= 1$
SM-MOEA [4]	The attenuation factor $\gamma = 0.1$
NMDE [15]	Scale factor $F = 0.5$, crossover rate $CR = 0.5$, $\theta = 0.6$, parameter $\psi = 0$
DMBDE	The threshold for removing features $\delta = 0.1 * \text{Score}_{\max}$, scale factor $F = 0.5$, crossover rate $CR = 0.5$

larger a value, e.g., 0.15 or 0.2, can help DMBDE achieve better HV performance on the high-dimensional datasets, the HV results tend to decrease on some low-dimensional datasets such as the Movement dataset. This is because some informative features might be removed on these datasets by using a slightly larger a value. Furthermore, $a = 0$ means only the irrelevant features in a dataset are removed. The overall HV results of $a = 0.05$ and $a = 0.1$ are better than that of $a = 0$. This indicates that removing some weakly relevant features can help DMBDE improve HV performance. Therefore, this study set a to 0.1.

5. Results

5.1. Overall Results

The detailed HV results of the eight methods on each test set are shown in Table 4. The results show that the proposed DMBDE method achieves the best on 10 out of the 14 datasets. There are only three losses of DMBDE in Table 4, and all happen in the low-dimensional datasets. One significant loss of DMBDE happens against SparseEA on the Arrhythmia dataset, and another two happen against DAEA on the Musk1 and Arrhythmia datasets. On the high-dimensional datasets apart from the Prostate dataset, the proposed DMBDE method shows the best performance.

In summary, DMBDE wins 75, draws 20 and loses 3 out of the 98 comparisons for the test HV results. Compared to the seven compared EMO algorithms, the non-dominated feature subsets of DMBDE have better distributions in the objective space.

5.2. Distributions of Non-dominated Fronts

To have an intuitive analysis, the distributions of the first non-dominated fronts with median run obtained by each algorithm are shown in Fig. 3 and Fig. 4. The first and second rows of Fig. 3 or Fig. 4 show all the non-dominated solutions on the training sets and test sets, respectively. In other words, all solutions shown in the second row are derived from the first row. Six datasets (the Sonar, Musk1, Madelon, 11Tumor, LungCancer, and 14Tumor datasets) are chosen since their non-dominated front distributions obtained by different algorithms are easier to be distinguished visually.

In Fig. 3, the feature subsets from the proposed DMBDE method show the lowest classification error rate on the Sonar dataset

Table 3
Average of training HV results of DMBDE with different a values

Dataset	0	0.05	0.1	0.15	0.2
WBCD	**0.953**±0.0	**0.953**±0.0	**0.953**±0.0	**0.953**±0.0	0.950±0.0
Sonar	0.896±0.011	**0.899**±0.010	0.897±0.011	0.898±0.012	0.887±0.004
Movement	**0.778**±0.010	0.769±0.005	0.764±0.005	0.740±0.005	0.700±0.007
Hillvally	**0.670**±0.002	0.668±0.004	**0.670**±0.002	0.669±0.003	0.657±0.003
Musk1	**0.993**±0.001	**0.993**±0.001	0.992±0.001	0.991±0.001	0.991±0.001
Multiple	**0.975**±0.001	**0.975**±0.001	**0.975**±0.001	**0.975**±0.001	**0.975**±0.001
Arrhythmia	**0.756**±0.005	0.755±0.004	**0.756**±0.005	0.755±0.004	0.755±0.004
Madelon	0.909±0.003	0.912±0.002	**0.914**±0.002	**0.914**±0.001	0.912±0.002
SRBCT	0.999±0.0	**1.0**±0.0	0.998±0.0	**1.0**±0.0	**1.0**±0.0
Leukemia	0.999±0.0	0.999±0.001	**1.0**±0.0	**1.0**±0.0	**1.0**±0.0
Prostate	0.991±0.007	0.994±0.006	**0.996**±0.005	**0.996**±0.006	**0.996**±0.006
11Tumor	0.938±0.011	0.942±0.010	0.942±0.011	0.950±0.009	**0.957**±0.009
LungCancer	0.990±0.004	0.988±0.004	0.990±0.004	**0.991**±0.004	**0.991**±0.005
14Tumor	0.673±0.013	0.669±0.009	0.668±0.013	0.675±0.010	**0.683**±0.010

Table 4
Average of test HV results of the eight methods

Dataset	SPEA2	MOEA/D-DE	GF-NSGAII	SparseEA	DAEA	SM-MOEA	NMDE	DMBDE
WBCD	0.913±0.009↓	0.896±0.022↓	0.916±0.005≈	0.910±0.011↓	0.914±0.009≈	0.904±0.009↓	0.915±0.003≈	**0.917**±0.001
Sonar	0.832±0.038↓	0.815±0.035↓	0.813±0.034↓	0.847±0.031≈	0.845±0.025≈	0.798±0.043↓	0.842±0.036≈	**0.850**±0.033
Movement	0.772±0.018↓	0.765±0.023↓	0.743±0.029↓	0.776±0.017↓	0.778±0.012↓	0.726±0.030↓	0.775±0.013↓	**0.798**±0.013
Hillvally	0.595±0.011≈	0.590±0.013≈	0.588±0.009↓	0.593±0.007≈	0.589±0.006↓	0.578±0.008↓	**0.598**±0.009≈	0.595±0.005
Musk1	0.965±0.010↓	0.963±0.008↓	0.956±0.007↓	0.971±0.003↓	**0.979**±0.002↑	0.961±0.005↓	0.973±0.007≈	0.975±0.004
Multiple	0.936±0.007↓	0.936±0.007↓	0.875±0.027↓	**0.951**±0.002≈	0.947±0.004↓	0.940±0.007↓	0.949±0.003≈	**0.951**±0.005
Arrhythmia	0.691±0.020≈	0.664±0.024↓	0.691±0.015≈	**0.732**±0.012↑	0.695±0.016↑	0.676±0.051≈	0.692±0.029≈	0.686±0.021
Madelon	0.883±0.010↓	0.872±0.011↓	0.891±0.004↓	**0.900**±0.006≈	0.891±0.005↓	0.892±0.012↓	0.894±0.008↓	**0.900**±0.004
SRBCT	0.926±0.019↓	0.834±0.008↓	0.910±0.016↓	0.918±0.037↓	0.959±0.034↓	0.907±0.048↓	0.981±0.022↓	**0.994**±0.014
Leukemia	0.782±0.048↓	0.702±0.024↓	0.792±0.036↓	0.934±0.036↓	0.949±0.041↓	0.920±0.059↓	0.972±0.037≈	**0.986**±0.019
Prostate	0.712±0.024↓	0.780±0.035↓	0.690±0.026↓	0.926±0.026↓	**0.937**±0.025≈	0.904±0.034↓	0.913±0.021↓	0.936±0.013
11Tumor	0.642±0.026↓	0.725±0.026↓	0.635±0.021↓	0.776±0.039↓	0.801±0.040↓	0.710±0.065↓	0.801±0.049↓	**0.859**±0.023
LungCancer	0.761±0.014↓	0.856±0.020↓	0.728±0.010↓	0.903±0.031↓	0.939±0.019↓	0.900±0.035↓	0.926±0.021↓	**0.946**±0.014
14Tumor	0.411±0.017↓	0.478±0.021↓	0.404±0.016↓	0.499±0.014↓	0.538±0.032↓	0.483±0.049↓	0.542±0.022↓	**0.594**±0.020
Sum	2≈, 12↓	1≈, 13↓	2≈, 12↓	1↑, 4≈, 9↓	2↑, 3≈, 9↓	1≈, 13↓	7≈, 7↓	N/A
Rank	5.68	6.61	6.61	3.36	3.00	6.21	2.82	1.71

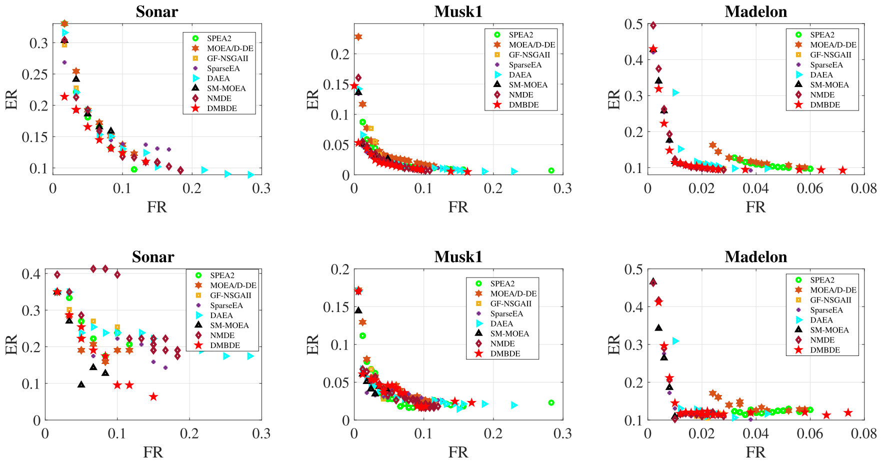

Fig. 3. Obtained PFs of eight algorithms on the training (the first row) and the test (the second row) sets.

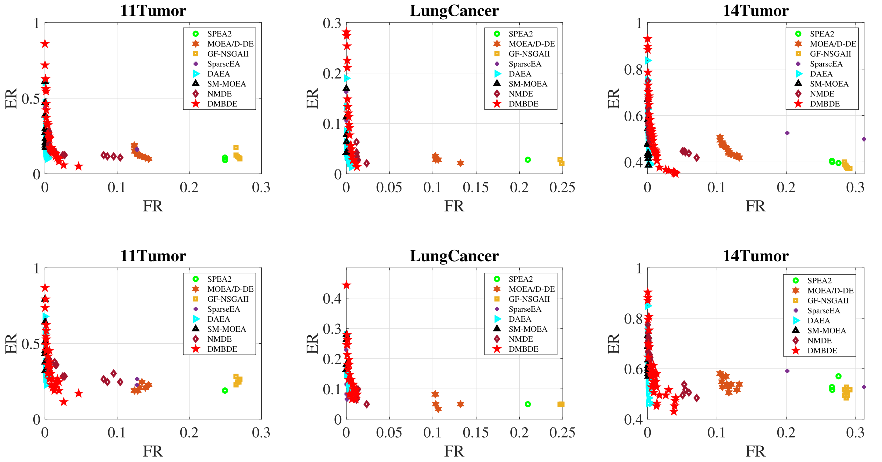

Fig. 4. Obtained PFs of eight algorithms on the training (the first row) and the test (the second row) sets.

among the eight methods. On the Musk1 dataset, although DAEA and SPEA2 find non-dominated feature subsets where the numbers of the selected features are over 2% of the original number (166) of features, these feature subsets show a slightly higher classification error rate. On the Madelon dataset, DMBDE can find non-dominated feature subsets where the number of the selected features over 6% of the original number (500) of features. However, the other seven methods fail to find those solutions. Furthermore, in Fig. 4, on the 11Tumor, LungCancer, and 14Tumor datasets, although different feature subsets from the eight algorithms can achieve similar the lowest training and test classification error rate, the feature subsets from DMBDE only contain no more than 5% of the original features. However, this number for SPEA2 and GF-NSGAII is over 20%. For MOEA/D-DE, this number is higher than 10%. Although the numbers of the selected features by SparseEA, DAEA, NMDE, and DMBDE are close, the feature subsets from the proposed DMBDE method have better distribution in the objective space.

In both the training and test sets, the non-dominated solutions of DMBDE are generally more diverse (reserving more Pareto-optimal solutions) and perform better (with fewer selected features and lower classification error rates) than the other methods.

5.3. Analysis on Feature Selection Performance

Generally, classification performance is preferred over the number of the selected features for classification problems. Among the multiple non-dominated feature subsets from one method, the feature subset with the lowest training classification error rate is chosen and its classification performance is evaluated on the test sets. Their average test classification accuracies (Ac), average numbers (d^*) of the selected features, and average F1 results obtained from the eight methods are presented in Tables 5–7, respectively. In addition, as a reference, the Ac and F1 performance of using all original features in a dataset are shown in the last column in Table 5 and Table 7, respectively.

Table 5
Average of test *Ac* results of the nine methods

Dataset	SPEA2	MOEA/D-DE	GF-NSGAII	SparseEA	DAEA	SM-MOEA	NMDE	DMBDE	All
WBCD	93.82±0.79↓	93.53±0.95↓	94.02±0.59≈	93.22±1.41↓	93.27±1.43↓	92.67±1.10↓	93.65±0.93↓	**94.15**±0.0	91.81
Sonar	80.21±5.85≈	79.02±4.29↓	79.50±4.05↓	83.02±4.69≈	81.96±4.03≈	78.15±4.27↓	82.75±5.71≈	81.87±4.85	**85.71**
Movement	75.83±2.44↓	76.88±2.81≈	73.77±3.41↓	76.48±2.26≈	76.70±1.82≈	71.74±3.56↓	75.65±2.11↓	**77.70**±2.31	73.15
Hillvally	55.49±1.34≈	54.03±1.32↓	54.33±1.23↓	54.62±1.52≈	54.24±1.27↓	53.78±0.92↓	**55.63**±1.34≈	55.24±1.11	51.10
Musk1	98.01±0.44↑	97.13±0.80↓	97.01±0.61↓	97.14±0.58↓	**98.14**±0.40↑	96.18±0.74↓	97.98±0.57↑	97.63±0.44	95.41
Multiple	96.24±0.54≈	96.32±0.4≈	88.58±4.09↓	96.64±0.49↑	96.46±0.55≈	94.83±1.11↓	96.38±0.55≈	96.17±0.67	**97.33**
Arrhythmia	64.44±2.17≈	62.38±2.62↓	65.18±1.41↑	**65.27**±2.14↑	64.58±1.61≈	62.72±4.56≈	63.86±2.62≈	64.28±1.55	54.41
Madelon	87.84±0.92≈	87.09±1.27↓	**88.66**±0.66↑	88.84±0.65↑	88.25±0.71≈	88.25±1.40≈	88.58±0.84≈	88.21±0.76	71.03
SRBCT	98.42±2.09≈	98.11±2.81≈	99.27±1.50≈	89.47±4.90↓	94.80±4.75↓	88.90±6.20↓	95.79±3.78↓	**99.86**±3.35	84.00
Leukemia	86.82±5.16↓	87.61±5.11↓	91.29±4.07↓	91.21±5.50↓	91.16±5.13↓	89.55±6.76↓	92.40±6.39≈	**95.14**±4.79	81.82
Prostate	86.21±3.56↓	85.03±4.84↓	85.99±3.39↓	90.11±3.81≈	**89.95**±4.12≈	88.60±4.31↓	86.76±4.03↓	89.25±3.34	87.10
11Tumor	80.94±3.85≈	77.58±3.93≈	79.69±3.20≈	73.58±4.82↓	73.90±4.36↓	66.16±6.69↓	76.86±5.70↓	**79.81**±4.25	69.81
LungCancer	**93.87**±1.65≈	92.49±2.54≈	93.66±1.25≈	88.91±3.56↓	91.15±2.45↓	88.09±3.83↓	91.11±2.98↓	92.79±2.63	93.44
14Tumor	48.28±2.71↓	46.56±2.39↓	47.81±2.69↓	43.19±2.98↓	45.84±4.65↓	41.83±5.64↓	47.06±3.85↓	**51.51**±3.11	44.09
Sum	1↑, 8≈, 5↓	5≈, 9↓	2↑, 4≈, 8↓	3↑, 4≈, 7↓	1↑, 6≈, 7↓	2≈, 12↓	1↑, 6≈, 7↓	N/A	
Rank	3.79	5.64	4.11	4.07	4.18	7.18	3.93	3.11	

Table 6
Average d^* results of the eight methods

Dataset	SPEA2	MOEA/D-DE	GF-NSGAII	SparseEA	DAEA	SM-MOEA	NMDE	DMBDE	All
WBCD	4.2±1.6↓	5.3±2.6↓	3.2±0.5≈	7.0±3.6↓	7.5±2.8↓	**2.1**±0.3↑	4.9±1.6↓	3.0±0.2	30
Sonar	8.9±1.9↑	10.8±4.5≈	6.2±1.3↑	11.8±3.8≈	13.6±4.1↓	**4.0**±0.9↑	9.1±2.4↑	10.6±3.0	60
Movement	14.5±6.1≈	14.1±5.4≈	6.1±0.8↑	10.9±3.7↑	15.9±6.1≈	**5.6**±0.8↑	12.9±5.2↑	15.5±3.6	90
Hillvally	10.4±2.2↑	9.9±3.4↑	8.3±1.9↑	9.3±2.1↑	10.7±3.4↑	**2.6**±0.6↑	11.0±1.7↑	15.5±4.5	100
Musk1	24.9±6.6↑	25.7±9.7↑	10.7±4.2↑	27.6±14.1↑	34.1±6.9≈	**8.2**±3.6↑	22.3±5.5↑	36.0±10.1	166
Multiple	67.3±13.4≈	80.2±18.9≈	**14.9**±9.8↑	95.9±19.3↓	84.4±11.9↓	35.7±10.9↑	83.1±12.1↓	73.0±15.9	240
Arrhythmia	12.6±2.8↑	18.6±9.7≈	11.4±2.2↑	13.2±3.8≈	23.9±9.0↓	**4.5**±0.9↑	11.1±3.1↑	15.2±4.2	279
Madelon	22.0±5.5≈	31.3±8.8↓	16.2±2.8↑	12.3±4.2↑	27.9±7.7↓	**5.9**±0.9↑	15.0±3.2↑	22.5±4.6	500
SRBCT	159.7±20.2↓	246.5±36.9↓	215.6±14.1↓	11.3±15.6≈	9.9±3.2≈	**5.6**±3.6↑	8.3±3.7↑	11.0±2.9	2,308
Leukemia	658.0±41.1↓	551.3±94.4↓	864.3±46.3↓	7.3±21.5↑	3.2±1.2↑	**3.1**±5.3↑	13.8±28.5≈	16.2±2.3	5,147
Prostate	2196.8±97.0↓	1285.3±167.9↓	2539.6±66.5↓	4.7±2.6↑	12.8±9.9↑	**2.8**±0.8↑	22.6±32.2↑	164.8±55.0	10,509
11Tumor	3233.4±230.1↓	1937.6±496.7↓	3464.8±94.1↓	1820.4±1381.1↓	71.8±56.0↑	**23.0**±48.6↑	855.5±442.9↓	482.0±128.9	12,533
LungCancer	2717.5±102.3↓	1496.2±378.6↓	3195.7±68.2↓	330.1±468.0≈	63.8±56.7↑	**6.0**±1.3↑	431.1±452.4↓	249.8±123.8	12,600
14Tumor	4058.8±202.9↓	2152.5±279.7↓	4384.6±40.4↓	1978.9±2523.8↓	42.9±18.9↑	**20.6**±9.7↑	783.7±399.0↓	548.4±133.0	15,009
Sum	4↑, 3≈, 7↓	2↑, 4≈, 8↓	7↑, 1≈, 6↓	6↑, 4≈, 4↓	6↑, 3≈, 5↓	14↑	8↑, 1≈, 5↓	N/A	
Rank	5.35	6.00	4.71	4.44	3.21	1.07	4.07	4.32	

Table 7
Average of test F1 results of the nine methods

Dataset	SPEA2	MOEA/D-DE	GF-NSGAII	SparseEA	DAEA	SM-MOEA	NMDE	DMBDE	All
WBCD	93.51±0.84↓	93.20±1.01↓	93.72±0.64≈	92.86±1.50↓	92.91±1.52↓	92.28±1.17↓	93.32±0.98↓	**93.86**±0.0	91.37
Sonar	79.75±6.10≈	78.64±4.39↓	79.20±4.04≈	82.63±4.79≈	81.67±4.10≈	77.65±4.45↓	82.41±5.82≈	81.56±4.90	**85.48**
Movement	75.79±2.98↓	77.63±2.77≈	73.34±3.82↓	76.63±2.86↓	76.85±1.94↓	71.40±3.56↓	75.38±2.80↓	**78.54**±2.31	75.12
Hillvally	55.48±1.34≈	54.00±1.32↓	54.30±1.22↓	54.61±1.53≈	54.23±1.27↓	53.74±0.92↓	**55.60**±1.34≈	55.23±1.10	51.10
Musk1	98.01±0.44↑	97.13±0.80↓	97.01±0.61↓	97.14±0.58↓	**98.14**±0.40↑	96.18±0.74↓	97.98±0.57↑	97.63±0.44	95.41
Multiple	96.17±0.55≈	96.27±0.40≈	88.57±4.06↓	96.59±0.50↑	96.41±0.55≈	94.81±1.14↓	96.33±0.57≈	96.10±0.69	**97.30**
Arrhythmia	64.75±3.93≈	61.00±4.35↓	**67.86**±1.98↑	64.64±3.56≈	64.90±3.53≈	62.59±5.96↓	63.32±3.68↓	65.68±4.14	61.19
Madelon	87.83±0.92≈	87.08±1.27↓	88.65±0.66↑	**88.83**±0.65↑	88.24±0.71≈	88.25±1.40≈	88.57±0.84≈	88.20±0.76	70.80
SRBCT	98.01±2.81≈	97.26±3.84≈	**98.96**±2.20≈	89.36±5.84↓	93.67±5.55↓	86.96±7.13↓	92.92±5.35↓	98.31±4.48	80.63
Leukemia	86.10±5.64↓	86.91±5.70↓	90.92±4.33↓	91.04±5.62↓	90.97±5.23↓	89.25±6.99↓	92.22±6.55≈	**95.05**±4.85	80.36
Prostate	85.71±3.71↓	84.73±4.90↓	85.53±3.46↓	**89.97**±3.82≈	89.82±4.12≈	88.46±4.34≈	86.66±4.04↓	89.13±3.38	86.75
11Tumor	75.78±5.23≈	73.27±5.73≈	74.35±4.21≈	67.64±5.49↓	70.28±5.87↓	60.62±8.61↓	73.95±6.65≈	**76.00**±5.24	66.24
LungCancer	92.19±2.21≈	89.87±4.63≈	91.20±3.63≈	83.66±7.15↓	88.49±3.47↓	82.70±7.72↓	88.18±4.60↓	90.81±3.67	**92.60**
14Tumor	37.60±4.08↓	36.40±3.27↓	37.65±3.06↓	35.21±2.57↓	37.10±4.92↓	33.08±5.01↓	36.56±4.36↓	**43.96**±3.55	35.47
Sum	1↑, 8≈, 5↓	5≈, 9↓	2↑, 5≈, 7↓	2↑, 4≈, 8↓	1↑, 5≈, 8↓	2≈, 12↓	1↑, 6≈, 7↓	N/A	
Rank	3.93	5.79	4.14	4.29	3.93	7.14	4.00	2.79	

Table 8
Average of test HV, Ac, and d^* results of the five methods

Datasets	MBDE-W			MBDE-M			MBDE-MF			NMBDE-MF			DMBDE		
	HV	Ac(%)	d^*	HV	Ac(%)	d^*	HV	Ac(%)	d^*	HV	Ac(%)	d^*	HV	Ac(%)	d^*
WBCD	**0.916**	94.07	3.1	**0.916**	**94.15**	**3.0**	**0.916**	**94.15**	**3.0**	**0.916**	**94.15**	**3.0**	**0.916**	**94.15**	**3.0**
Sonar	0.830	81.77	**8.8**	0.836	81.26	9.4	0.840	81.30	9.1	0.840	81.68	11.3	**0.850**	**81.87**	10.6
Movement	0.774	76.16	**10.7**	0.774	76.39	11.5	0.797	77.21	11.8	**0.799**	**78.02**	14.9	0.798	77.70	15.5
Hillvally	0.593	55.11	**12.2**	0.593	54.92	12.8	0.595	54.98	13.1	**0.597**	**55.25**	15.6	0.595	55.24	15.5
Musk1	0.948	**98.21**	**21.8**	0.950	**98.21**	22.3	0.948	97.77	25.1	0.969	97.63	41.5	**0.975**	97.63	36.0
Multiple	0.910	96.29	68.2	0.906	96.20	69.1	0.908	96.29	**65.4**	**0.952**	**96.51**	83.6	0.951	96.17	73.0
Arrhythmia	0.684	64.25	17.2	0.667	**64.34**	15.4	0.689	64.31	**14.7**	**0.697**	64.08	15.5	0.686	64.28	15.2
Madelon	0.883	87.98	30.8	0.877	87.99	28.1	0.886	88.12	24.4	**0.903**	88.07	25.4	0.900	**88.21**	**22.5**
SRBCT	0.983	97.50	18.1	0.986	97.05	**6.4**	0.990	98.53	11.6	0.988	97.49	11.8	**0.994**	**99.86**	11.0
Leukemia	0.978	95.23	39.0	0.978	96.62	40.2	**0.996**	**98.38**	**15.6**	0.981	96.44	16.2	0.986	95.14	16.2
Prostate	0.935	87.83	119.0	0.936	88.23	123.4	0.931	87.53	**98.2**	**0.941**	**89.57**	178.8	0.936	89.25	164.8
11Tumor	0.823	72.58	203.3	0.830	75.97	203.4	0.829	74.59	**177.5**	0.852	78.81	551.6	**0.859**	**79.81**	482.0
LungCancer	0.935	91.31	153.4	0.936	90.98	165.1	0.935	90.90	**134.6**	0.941	92.35	231.6	**0.946**	**92.79**	249.8
14Tumor	0.560	48.85	229.8	0.563	47.56	**221.0**	0.562	46.92	215.8	0.586	51.22	600.2	**0.594**	**51.51**	548.4
Rank	4.32	3.57	2.79	3.86	3.43	2.75	3.21	3.14	1.82	1.64	2.50	4.21	1.63	2.28	3.43

5.3.1. Classification Accuracy and Dimensionality Reduction Analysis

In Table 6, the number of features selected by the proposed DMBDE method is one to two orders of magnitude lower than the original size. The features selected by DMBDE achieve better *Ac* results than using all features with an increase of more than 4% on eight datasets. On the Madelon, SRBCT, Leukemia, and 11Tumor datasets, feature subsets from DMBDE obtained 10% higher accuracy than All on average.

On two datasets (Sonar and Multiple) in Table 6, although using all features achieves the highest *Ac* value among the nine methods, DMBDE is the overall best. Furthermore, in Table 6, DMBDE selects fewer features than SPEA2, MOEA/D-DE, and GF-NSGAII, especially on the high-dimensional datasets. The highest dimensionality reduction can be seen on the Leukemia dataset where DMBDE selects 32 times fewer features than the three methods (SPEA2, MOEA/D-DE, and GF-NSGAII) and still can achieve the highest *Ac* value. Although SM-MOEA shows the best d^* performance among the eight methods, its accuracy results are unsatisfactory. For example, on the 14Tumor dataset, SM-MOEA selects only 20 features but gets the lowest *Ac* value. In SM-MOEA, a feature will be deleted in the nonelite individuals if the feature in elite individuals is not selected, thereby leading SM-MOEA to select fewer features. In addition, SparseEA, DAEA, and NMDE show similar or better d^* results than DMBDE, but DMBDE has the best *Ac* performance. On the Prostate dataset, both SparseEA and DAEA select less than 15 features achieving similar *Ac* performance with DMBDE, but DMBDE selects over 160 features. On the two datasets (11Tumor and 14Tumor), DMBDE selects fewer features and achieves better accuracy than SparseEA and NMDE.

In summary, DMBDE wins 90, draws 51 and loses 55 out of the 196 comparisons in terms of the test *Ac* and d^* results. The results show that DMBDE can generally achieve promising feature selection performance in classification.

5.3.2. F1 Score Analysis

In Table 7, the proposed DMBDE method shows the overall best performance in terms of F1 score. Specifically, DMBDE obtains the largest F1 value and the second largest F1 score on five (WBCD, Movement, Leukemia, 11Tumor, and 14Tumor) and three (Arrhythmia, SRBCT, and LungCancer) datasets, respectively. On the Sonar, Multiple, and LungCancer datasets, using all features achieves the best F1 performance among the nine methods (slightly better F1 score than DMBDE).

In summary, DMBDE wins 56, draws 35, and loses 7 out of the 98 comparisons in terms of the test F1 results.

5.4. Analysis on Key Strategies

DMBDE has three major components: the feature removal strategy, the binary mutation operator, and the diversity-based environmental selection operator. It is important to discuss their respective effectiveness. Four variant algorithms of DMBDE are made to show the effectiveness of the three components:

1) MBDE-W selects informative features from the original whole feature set. When performing mutation, three vectors will be randomly selected from the whole population P in Eqs. (10) and (11). In other words, no neighborhood information is used. The environmental selection process of MBDE-W is the same as that of NSGA-II [33];
2) MBDE-M removes irrelevant features and weakly relevant features only based on MIC. That means the score of a feature will be obtained by $Sc(F_j) = \text{MIC}(F_j)$ rather than using Eq. (8). The remaining parts of MBDE-M such as the mutation operator and the environmental selection process are the same as MBDE-W;
3) MBDE-MF differs from MBDE-M only in the removal process. MBDE-MF removes irrelevant features and weakly relevant features followed by Eq. (8). Both MIC and Fisher scores are used;
4) NMBDE-MF is formed by adding the developed mutation operator in Section 3.3 to MBDE-MF. That means the neighborhood information is considered when updating the population.

One note is that the only difference between MNBDE-MF and DMBDE is that DMBDE uses the proposed diversity-based selection operator, while MNBDE-MF does not. The HV, *Ac*, and d^* results of the five algorithms are shown in Table 8.

5.4.1. Analysis on Feature Removal Operator

In Table 8, compared with MBDE-W, MBDE-M can maintain or even improve the HV and *Ac* performance on most of the used datasets. The d^* performance between the two methods is similar. The comparison between MBDE-W and MBDE-M shows that using MIC to remove features can help the proposed method slightly improve the final feature selection performance. In comparison with

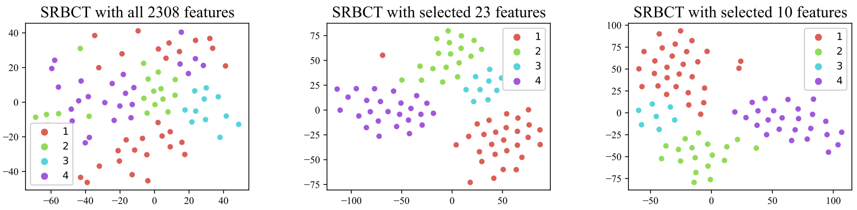

Fig. 5. Visualization of instances' distributions on the SRBCT dataset.

MBDE-M, MBDE-MF achieves significantly better HV, accuracy, and subset size performance. On the Leukemia dataset, the feature subset obtained from MBDE-MF includes the lowest number of features and the highest classification accuracy among the five methods.

To have a further intuitive analysis, Fig. 5 visualizes the distribution of the instances using t-SNE [49] on the SRBCT dataset. In Fig. 5, the input of the figure on the left is the whole SRBCT feature set. The inputs of the figures on the middle and right are the feature subset with the lowest training error rate in the obtained non-dominated solutions from the median run of MBDE-W and MBDE-MF, respectively. In Fig. 5, the distribution of data points using all features is the situation where the instances belonging to different classes are overlapped. However, the data point distributions with the selected features from MBDE-W and MBDE-MF are quite different from the original features. In the middle figure (23 features selected by MBDE-W) of Fig. 5, most instances with the same class label tend to form homogeneous clusters. Furthermore, in the right figure (10 features selected by MBDE-MF) of Fig. 5, all instances are perfectly separated by the selected features.

The results show that simultaneously using MIC and Fisher criterion to remove features can help the proposed method achieve better feature selection performance.

5.4.2. *Analysis on Niching-based Mutation*

Compared with MBDE-MF, NMBDE-MF increases HV value on 10 datasets and improves *Ac* performance on 8 datasets but selects more features on 13 datasets. On the last four high-dimensional datasets (Prostate, 11Tumor, LungCancer and 14Tumor), NMBDE-MF obtains 2% higher average accuracy than MBDE-MF. The comparison between MBDE-MF and NMBDE-MF shows that the used niching-based mutation operator can produce better feature subsets during the evolutionary training process.

5.4.3. *Analysis on Environmental Selection Operator*

The proposed diversity-based environmental selection operator focuses on the diversity or dissimilarity of feature subsets. As shown in Table 8, compared with NMBDE-MF, DMBDE significantly improves the *Ac* and d^* results on most of the tested datasets. On the Sonar, Arrhythmia, Madelon, SRBCT, 11Tumor, and 14Tumor datasets, both the *Ac* and d^* performance are enhanced. This shows that picking up solutions with larger diversity scores tends to obtain better feature selection performance. Additionally, since the crowding distance is considered the second selection criterion in DMBDE, the HV performance between NMBDE-MF and DMBDE is similar. The comparison between NMBDE-MF and DMBDE shows that the proposed diversity-based environmental selection operator can further improve the classification accuracy by increasing the population diversity.

5.5. *Training Time Analysis*

All experiments in this paper use the Mahuika High-Performance Computing (HPC) cluster of the New Zealand eScience Infrastructure (NeSI) [50]. Specifically, for each run of one algorithm, a dual-core CPU processor with 5GB RAM in each CPU is adopted. For

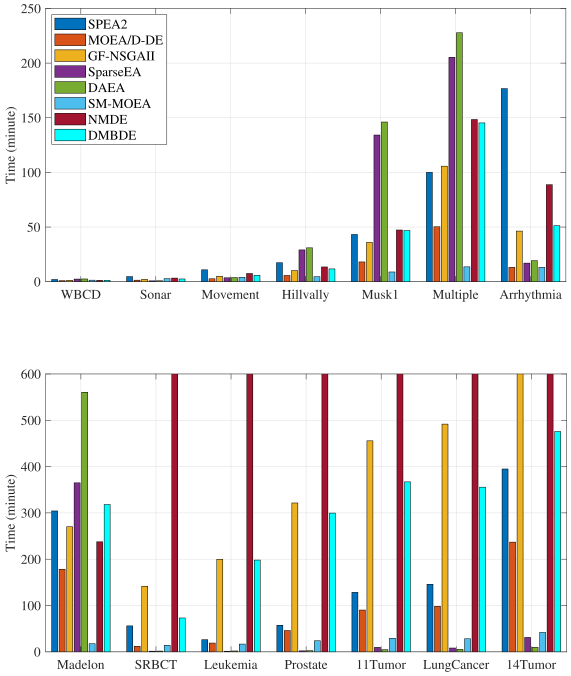

Fig. 6. CPU times of different feature selection algorithms on the evolutionary training process.

an intuitive analysis, the average computational times of the algorithms for the 14 datasets in minutes are divided into two figures, as shown in Fig. 6.

In Fig. 6, the eight methods consume a short time (less than 20 minutes) on the WBCD, Sonar, and Movement datasets. On four datasets with a large number of instances including Hillvally, Musk1, Multiple, and Madelon, DAEA spends the longest time. On the last six high-dimensional datasets, the two slowest algorithms among the eight methods are NMDE and GF-NSGAII. For instance, the average training times (over 600 minutes) of NMDE and GF-NSGAII on the 14Tumor dataset are more than six times longer than that of SparseEA, DAEA, and SM-MOEA (less than 100 minutes). This is because of the complex environmental selection process from both NMDE and GF-NSGAII.

On the large datasets representing harder feature selection tasks, the overall training time of DMBDE is between 1.2 and 7.9 hours. Although DMBDE finishes the evolutionary training process in a longer time than SPEA2, MOEA/D-DE, SparseEA, SM-MOEA, and DAEA on most of the used high-dimensional datasets, DMBDE obtains better feature selection performance.

5.6. Further Analysis on Different Learning Algorithms

To explore the effect of different learning algorithms, the proposed DMBDE method is combined with two other learning algorithms, i.e., Logistic Regression (LR) and Random Forest (RF). The averages of test HV results, test classification accuracies (*Ac*), test F1 scores, and the numbers (d^*) of the selected features, from the three methods, are shown in Table 9.

In Table 9, it is obvious that by using different learning algorithms, the obtained feature subsets by DMBDE can have different performances. Although DMBDE with LR classifier can have satisfactory classification performance on the Hillvally dataset, it achieves a quite low test classification accuracy on the Madelon dataset. Conversely, DMBDE with RF or KNN obtain low performance on the Hillvally dataset, but it has significantly better performances than DMBDE with LR on the Madelon dataset in terms of the HV, *Ac*, and F1 score indicators. These situations can happen due to many reasons. Complicated feature interactions may cause optimal feature subsets to change when different learning algorithms are employed. Different learning algorithms are good at dealing with different classification tasks. For example, some features in the Hillvally dataset might have linear relationships with the class labels, thus LR achieves better performance.

Although the classification accuracy or the F1 score might be unsatisfactory on some datasets such as the 14Tumor dataset, they are used as challenging feature selection problems, to motivate the development of better algorithms in this research area. More importantly, the results show that the proposed DMBDE method can reduce the dimensionality while maintaining or improving the classification accuracy and F1 score although using different learning algorithms. For example, in Table 9, DMBDE with LR uses no more than six features that can also achieve a good classification accuracy and F1 score (nearly 100%).

6. Conclusions

The goal of this paper was to design a new method for multi-objective feature selection tasks. The goal has been successfully achieved by proposing a new measure to remove the irrelevant and weakly relevant features in a dataset, developing a new binary mutation operator in DE, and designing a diversity-based environmental selection mechanism. The proposed DMBDE method was compared with seven multi-objective feature selection algorithms on 14 datasets. The results indicated that DMBDE outperformed the competing algorithms in terms of the HV, F1 score, and classification accuracy, and successfully selected a much smaller number of features while achieving better classification accuracy than most of the compared algorithms. Further analyses showed that all three new components (the feature removal operator, the developed mutation operator, and the diversity-based environmental selection mechanism) contributed positively to enhancing the performance of the proposed method.

Further research in this direction could include investigating a more efficient strategy that can further reduce the subset size of the obtained solutions and using different representations of solutions to better explore the solution space, especially on high-dimensional datasets. These directions are currently pursued by the authors. By using feature construction methods, sampling techniques, or data augmentation approaches, the classification performance might be further improved on some challenging feature selection problems. The authors will also explore these research topics in the future.

CRediT Authorship Contribution Statement

Peng Wang: Conceptualization, Methodology, Writing-original draft, Software. **Bing Xue**: Supervision, Writing-review & editing, Funding acquisition. **Jing Liang**: Supervision, Writing-review & editing, Project administration, Funding acquisition. **Mengjie Zhang**: Supervision, Writing-review & editing, Project administration, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Table 9
Average of test HV, *Ac*, F1, and d^* results of the three methods

Datasets	DMBDE (LR)				DMBDE (RF)				DMBDE (KNN)			
	HV	*Ac*(%)	F1	d^*	HV	*Ac*(%)	F1	d^*	HV	*Ac*(%)	F1	d^*
WBCD	0.924↑	94.04≈	93.74≈	7.9	**0.927**↑	**95.46**↑	**95.25**↑	6.1	0.917	94.15	93.86	**3.0**
Sonar	0.810↓	80.06↓	79.69↓	**9.4**	0.847≈	**83.02**≈	**82.85**≈	12.2	**0.850**	81.87	81.56	10.6
Movement	0.664↓	65.99↓	64.21↓	24.7	0.766↓	74.29↓	74.99↓	17.7	**0.798**	**77.70**	**78.54**	**15.5**
Hillvally	**0.984**↑	**99.98**↑	**99.98**↑	**5.7**	0.619↑	57.94↑	57.91↑	10.6	0.595	55.24	55.23	15.5
Musk1	0.972≈	98.25↑	98.25↑	62.6	**0.986**↑	**99.05**↑	**99.05**↑	**19.6**	0.975	97.63	97.63	36.0
Multiple	0.932↓	94.36↓	94.27↓	**69.8**	0.950≈	**96.20**≈	**96.19**≈	72.3	**0.951**	96.17	96.10	73.0
Arrhythmia	0.713↑	65.56≈	**65.89**≈	30.2	**0.766**↑	**72.52**↑	64.73≈	16.9	0.686	64.28	65.68	**15.2**
Madelon	0.646↓	57.45↓	57.44↓	65.2	0.894↓	88.09≈	88.09≈	**12.9**	**0.900**	**88.21**	**88.20**	22.5
SRBCT	0.993≈	97.83↓	96.82↓	**5.2**	0.902↓	86.08↓	83.74↓	5.7	**0.994**	**99.86**	**98.31**	11.0
Leukemia	0.957↓	89.42↓	89.08↓	**12.2**	0.957↓	91.89↓	91.73↓	15.3	**0.986**	**95.14**	**95.05**	16.2
Prostate	**0.943**↑	90.05≈	89.92≈	**71.4**	0.942↑	**92.04**↑	**91.88**↑	86.3	0.936	89.25	89.13	164.8
11Tumor	**0.924**↑	**88.11**↑	**84.92**↑	422.9	0.908↑	85.97↑	79.93↑	**277.6**	0.859	79.81	76.00	482.0
LungCancer	**0.949**≈	91.09↓	86.27↓	**143.8**	0.934↓	90.93↓	88.07↓	208.3	0.946	**92.79**	**90.81**	249.8
14Tumor	**0.696**↑	**62.04**↑	**55.74**↑	946.0	0.615↑	56.56↑	50.54↑	**394.7**	0.594	51.51	43.96	548.4
Sum	6↑, 3≈, 5↓	4↑, 3≈, 7↓	4↑, 3≈, 7↓		7↑, 2≈, 5↓	7↑, 3≈, 4↓	6↑, 4≈, 4↓		N/A			

Acknowledgements

This work was supported in part by the Marsden Fund of New Zealand Government under Contracts VUW1913 and VUW1914, National Natural Science Foundation of China (NSFC) under Grant 61922072 and 61876169, the Science for Technological Innovation Challenge (SfTI) fund under grant E3603/2903, MBIE Data Science SSIF Fund under the contract RTVU1914, MBIE Research Program under MBIE C11X2001 and FPR Contract 39186, China Scholarship Council (CSC) and Victoria University Joint Scholarship.

References

[1] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, Journ, Mach. Learn. Research 3 (Mar) (2003) 1157–1182.

[2] Q. Al-Tashi, S.J. Abdulkadir, H.M. Rais, S. Mirjalili, H. Alhussian, Approaches to multi-objective feature selection: A systematic literature review, IEEE Access 8 (2020) 125076–125096.

[3] B. Xue, M. Zhang, W.N. Browne, X. Yao, A survey on evolutionary computation approaches to feature selection, IEEE Trans. Evol. Comput. 20 (4) (2015) 606–626.

[4] F. Cheng, F. Chu, Y. Xu, L. Zhang, A steering-matrix-based multiobjective evolutionary algorithm for high-dimensional feature selection, IEEE Trans. Cybern.

[5] D. Whitley, A genetic algorithm tutorial, Statist. Comp. 4 (2) (1994) 65–85.

[6] I.S. Oh, J.S. Lee, B.R. Moon, Hybrid genetic algorithms for feature selection, IEEE Trans. Patt. Analys. Mach. Intell. 26 (11) (2004) 1424–1437.

[7] A. Li, B. Xue, M. Zhang, Multi-objective feature selection using hybridization of a genetic algorithm and direct multisearch for key quality characteristic selection, Inf. Sci. 523 (2020) 245–265.

[8] R. Storn, K. Price, Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, Journ. Global Optimiz. 11 (4) (1997) 341–359.

[9] P. Wang, B. Xue, J. Liang, M. Zhang, Multiobjective differential evolution for feature selection in classification, IEEE Trans. Cybern.

[10] O. Tarkhaneh, T.T. Nguyen, S. Mazaheri, A novel wrapper-based feature subset selection method using modified binary differential evolution algorithm, Inf. Sci. 565 (2021) 278–305.

[11] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Intern. Conf. Neural Networks, Vol. 4, IEEE, 1995, pp. 1942–1948.

[12] X. Song, Y. Zhang, Y. Guo, X. Sun, Y. Wang, Variable-size cooperative coevolutionary particle swarm optimization for feature selection on high-dimensional data, IEEE Trans. Evol. Comput. 24 (5) (2020) 882–895.

[13] J. Ma, X. Gao, A filter-based feature construction and feature selection approach for classification using genetic programming, Knowledge-Based Syst. 196 (2020), 105806.

[14] Y. Zhang, D. Gong, X. Gao, T. Tian, X. Sun, Binary differential evolution with self-learning for multi-objective feature selection, Inf. Sci. 507 (2020) 67–85.

[15] P. Wang, B. Xue, J. Liang, M. Zhang, Differential evolution based feature selection: A niching-based multi-objective approach, IEEE Trans. Evol. Comput.

[16] K. Chen, B. Xue, M. Zhang, F. Zhou, Correlation-guided updating strategy for feature selection in classification with surrogate-assisted particle swarm optimisation, IEEE Trans. Evol. Comput.

[17] H. Xu, B. Xue, M. Zhang, A duplication analysis based evolutionary algorithm for bi-objective feature selection, IEEE Trans. Evol. Comput. 25 (2) (2020) 205–218.

[18] F. Cheng, J.J. Cui, Q.J. Wang, L. Zhang, A variable granularity search based multi-objective feature selection algorithm for high-dimensional data classification, IEEE Trans. Evol. Comput.

[19] C. Yue, J. Liang, B. Qu, K. Yu, H. Song, Multimodal multiobjective optimization in feature selection, in: IEEE Congr. Evol. Comput., 2019, pp. 302–309.

[20] L. Yu, H. Liu, Feature selection for high-dimensional data: A fast correlation-based filter solution, in: Intern. Conf. Mach. Learn., 2003, pp. 856–863.

[21] X. Song, Y. Zhang, D. Gong, X. Gao, A fast hybrid feature selection based on correlation-guided clustering and particle swarm optimization for high-dimensional data, IEEE Trans. Cybern.

[22] P. Viola, W.M. Wells III, Alignment by maximization of mutual information, Intern. Journ. Computer Vision 24 (2) (1997) 137–154.

[23] X. Lin, C. Li, W. Ren, X. Luo, Y. Qi, A new feature selection method based on symmetrical uncertainty and interaction gain, Comput. Biol. Chemistry 83 (2019), 107149.

[24] D.N. Reshef, Y.A. Reshef, H.K. Finucane, S.R. Grossman, G. McVean, P.J. Turnbaugh, E.S. Lander, M. Mitzenmacher, P.C. Sabeti, Detecting novel associations in large data sets, Science 334 (6062) (2011) 1518–1524.

[25] Y. Zhang, S. Jia, H. Huang, J. Qiu, C. Zhou, A novel algorithm for the precise calculation of the maximal information coefficient, Scient. Reports 4 (1) (2014) 1–5.

[26] P.E. Hart, D.G. Stork, R.O. Duda, Pattern classification, Wiley Hoboken, 2000.

[27] I. Kononenko, Estimating attributes: Analysis and extensions of relief, Europ. Conf. Mach. Learn., Springer, in, 1994, pp. 171–182.

[28] H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Patt. Analys. Mach. Intell. 27 (8) (2005) 1226–1238.

[29] D. López, S. Ramírez-Gallego, S. García, N. Xiong, F. Herrera, BELIEF: A distance-based redundancy-proof feature selection method for big data, Inf. Sci. 558 (2021) 124–139.

[30] J. Che, Y. Yang, L. Li, X. Bai, S. Zhang, C. Deng, Maximum relevance minimum common redundancy feature selection for nonlinear data, Inf. Sci. 409 (2017) 68–86.

[31] S. Salesi, G. Cosma, M. Mavrovouniotis, Taga: Tabu asexual genetic algorithm embedded in a filter/filter feature selection approach for high-dimensional data, Inf. Sci. 565 (2021) 105–127.

[32] E. Hancer, B. Xue, M. Zhang, Fuzzy filter cost-sensitive feature selection with differential evolution, Knowledge-Based Syst. 241 (2022), 108259.

[33] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197.

[34] T. Li, Z. Zhan, J. Xu, Q. Yang, Y. Ma, A binary individual search strategy-based bi-objective evolutionary algorithm for high-dimensional feature selection, Inf. Sci. 610 (2022) 651–673.

[35] A. Telikani, A. Tahmassebi, W. Banzhaf, A.H. Gandomi, Evolutionary machine learning: A survey, ACM Comp. Surveys 54 (8) (2021) 1–35.

[36] B.H. Nguyen, B. Xue, M. Zhang, A survey on swarm intelligence approaches to feature selection in data mining, Swarm Evol. Comput. 54 (2020), 100663.

[37] R.N. Khushaba, A. Al-Ani, A. Al-Jumaily, Feature subset selection using differential evolution and a statistical repair mechanism, Expert Syst. Appl. 38 (9) (2011) 11515–11526.

[38] P. Wang, B. Xue, J. Liang, M. Zhang, Differential evolution with duplication analysis for feature selection in classification, IEEE Trans. Cybern.

[39] F. Pukelsheim, The three sigma rule, Amer. Statist. 48 (2) (1994) 88–91.

[40] Z. Wang, Y. Zhou, J. Zhang, Adaptive estimation distribution distributed differential evolution for multimodal optimization problems, IEEE Trans. Cybern.

[41] D. Dua, C. Graff, UCI machine learning repository (2017). http://archive.ics.uci.edu/ml.

[42] B. Xue, M. Zhang, W.N. Browne, A comprehensive comparison on evolutionary feature selection approaches to classification, Intern. Journ. Computat. Intell. Appl. 14 (02) (2015) 1550008.

[43] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: Improving the strength pareto evolutionary algorithm, TIK-report 103.

[44] Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput. 11 (6) (2007) 712–731.

[45] P. Wang, B. Xue, M. Zhang, J. Liang, A grid-dominance based multi-objective algorithm for feature selection in classification, in: IEEE Congr. Evol. Comput., 2021, pp. 2053–2060.

[46] Y. Tian, X. Zhang, C. Wang, Y. Jin, An evolutionary algorithm for large-scale sparse multiobjective optimization problems, IEEE Trans. Evol. Comput. 24 (2) (2019) 380–393.

[47] E. Zitzler, L. Thiele, M. Laumanns, C.M. Fonseca, V.G. Da Fonseca, Performance assessment of multiobjective optimizers: An analysis and review, IEEE Trans. Evol. Comput. 7 (2) (2003) 117–132.

[48] R. Jiao, B. Xue, M. Zhang, Solving multi-objective feature selection problems in classification via problem reformulation and duplication handling, IEEE Trans. Evol. Comput.

[49] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, Journ. Mach. Learn. Research 9 (11).

[50] A. Pletzer, W. Hayek, C. Scott, B. Corrie, G. Rae, How NeSI helps users run better and faster on new zealand's supercomputing platforms, IEEE Intern. Conf. e-Science, in, 2017, pp. 465–466.