Knowledge-Based Systems 140 (2018) 103–119

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Differential evolution for filter feature selection based on information theory and feature ranking

Emrah Hancer[a,b], Bing Xue[b,*], Mengjie Zhang[b]

[a] Department of Computer Technology and Information Systems, Mehmet Akif Ersoy University, 15039 Burdur, Turkey
[b] Evolutionary Computation Research Group, School of Engineering and Computer Science, Victoria University of Wellington, Wellington 6140, New Zealand

ARTICLE INFO

Article history:
Received 19 May 2016
Revised 22 October 2017
Accepted 24 October 2017
Available online 2 November 2017

Keywords:
Mutual information
ReliefF
Fisher Score
differential evolution
feature selection

ABSTRACT

Feature selection is an essential step in various tasks, where filter feature selection algorithms are increasingly attractive due to their simplicity and fast speed. A common filter is to use mutual information to estimate the relationships between each feature and the class labels (mutual relevancy), and between each pair of features (mutual redundancy). This strategy has gained popularity resulting a variety of criteria based on mutual information. Other well-known strategies are to order each feature based on the nearest neighbor distance as in ReliefF, and based on the between-class variance and the within-class variance as in Fisher Score. However, each strategy comes with its own advantages and disadvantages. This paper proposes a new filter criterion inspired by the concepts of mutual information, ReliefF and Fisher Score. Instead of using mutual redundancy, the proposed criterion tries to choose the highest ranked features determined by ReliefF and Fisher Score while providing the mutual relevance between features and the class labels. Based on the proposed criterion, two new differential evolution (DE) based filter approaches are developed. While the former uses the proposed criterion as a single objective problem in a weighted manner, the latter considers the proposed criterion in a multi-objective design. Moreover, a well known mutual information feature selection approach (MIFS) based on maximum-relevance and minimum-redundancy is also adopted in single-objective and multi-objective DE algorithms for feature selection. The results show that the proposed criterion outperforms MIFS in both single objective and multi-objective DE frameworks. The results also indicate that considering feature selection as a multi-objective problem can generally provide better performance in terms of the feature subset size and the classification accuracy.

1. Introduction

Classification is typically referred as a supervised learning task in machine learning that infers a relationship between features (characteristics of the dataset) and the class labels. However, the presence of a large number of features often leads to challenges such as overfitting, high computational complexity and low interpretability of the final model [1]. One reason for this is widely known as the curse of dimensionality that arises according to the ratio between the number of features and the number of instances. The most common way to alleviate such problems is to reduce the number of features under consideration using either feature construction or feature selection [1,2].

Feature construction aims to transform the dataset from the high dimensional space to a lower dimensional space by combining the original low-level features to a small number of high-level features, which is better suited for learning process. However, feature construction cannot be easily interpreted since the physical meaning of the original features cannot be retrieved. Feature selection aims to choose a feature subset from the available original features of a dataset, which better contributes to the learning process. In other words, the aim of feature selection is to discard features that are detrimental to the subsequent learning process [3,4]. Feature selection approaches can be categorized into wrappers, embedded and filters based on the evaluation criteria [5]. Wrappers use a learning algorithm (classifier or regression) as a part of evaluation to measure the goodness of the chosen feature subset. Although wrappers are among the most preferred feature selection approaches, there are at least four drawbacks [6]: 1) high computational complexity, 2) the optimal feature subset for a learner may not be optimal for a different learner, 3) determining the user-specified parameters of the learner may be time consuming, and 4) inherent learner limitations (e.g. some learners cannot deal with multi-class classification). Embedded approaches incorporate

[*] Corresponding author.
E-mail address: bing.xue@ecs.vuw.ac.nz (B. Xue).

https://doi.org/10.1016/j.knosys.2017.10.028
0950-7051/© 2017 Elsevier B.V. All rights reserved.

knowledge about the specific structure of the classification algorithm used by a certain learning algorithm. Embedded approaches are computationally less intensive than wrappers. However, they still have high computational complexity and the selected feature subset is dependent on the learning algorithm. Due to these limitations, we specifically focus on filters in this study. Wrapper and embedded approaches are not the focus of this paper and will not be further discussed here. Recent works on wrappers and embedded approaches can be found in [5,7–12].

Filters evaluate feature subsets based on some predefined metrics or information content (e.g. statistical tests) instead of using the learners, i.e., there exists no dependence between the learner (or classifier) and the selected features. Accordingly, filters are more general than wrapper and embedded approaches. In the literature, there have been a wide range of criteria and metrics used for the evaluation of feature subsets such as inconsistency rate, inference correlation, fractal dimension, distance measure and mutual information. Among them, mutual information can be treated as the most preferred and widely investigated for filters due to two main properties [6]: 1) measuring different kinds of relationship between random variables and 2) preserving stability under transformations in the feature space that are invertible and differentiable. Based on mutual information, Battiti [13] proposed the mutual information feature selection (MIFS) method including three fundamental points: 1) features are categorized as relevant and redundant; 2) an heuristic function is used to select features controlling the tradeoff between relevance and redundancy; and 3) a greedy search is applied. Other representative examples of mutual information based approaches are maximum relevance and minimum redundancy (mRmR) [14], uniformly improved MIFS (MIFS-U) [15], and conditional mutual information maximization (CMIM) [16]. Although they are simple to implement and reduce the feature subset size, a selected feature cannot be later removed or changed due to their static greedy search mechanism.

To address these problems, researchers have tried to design mutual information based filter approaches with evolutionary computation (EC) techniques such as particle swarm optimization (PSO) [17], genetic algorithms (GAs) [18], ant colony optimization (ACO) [19] and differential evolution (DE) [20] due to their global search ability. Besides such representative ones, recently developed EC techniques such as artificial bee colony [21], and bacterial colony optimization [22] have also been investigated to obtain better feature subsets for the classification.

However, the potential of EC for feature selection has not been fully investigated. For example, filter based approaches are often computationally cheap, but there is much less work on filters than on wrappers because the fitness functions based on filters are more difficult to design. The most widely used filter measure is mutual information. Although EC with mutual information has achieved better results than classical greedy search, most of such methods just directly adopted existing heuristic/fitness functions as the objective without significant or major improvement, which may limits their performance [5]. Furthermore, although feature selection can be considered as a multi-objective problem, there are only a few works on multi-objective filter feature selection [5,23]. Developing good filter based feature selection methods is still an open issue.

Among EC methods, DE is a relatively recent but highly popular approach. As pointed in [24], DE has been proven to be better than other EC methods in a wide range of problems. Compared to most other EC methods, DE is also much simpler and straightforward to implement, which allows practitioners from other fields, who may not be experts in programming, to implement and tune it to solve the domain-specific problem. Furthermore, DE only has a few parameters to control and the space complexity is low as well. These are particularly important for feature selection since it is a multi-disciplinary area involving researchers from many different fields, but work on DE for feature selection is much less than other EC methods, e.g. GAs and PSO [5]. Furthermore, feature selection is essentially a multi-objective approach, maximizing the classification accuracy and minimizing the number of features [25]. EC methods are particularly good for solving multi-objective problems since their population based mechanism can produce multiple trade-off solutions in a single run [26]. Despite the superior performance of multi-objective DE, there has been almost no work exploring the potential of DE for multi-objective filter feature selection.

1.1. Goals

The overall goal of this paper is to develop filter based feature selection approaches based on information theory, feature ranking and EC techniques to search for a set of non-dominated solutions (feature subsets) yielding a smaller number of features and a similar or even better classification performance on the K-nearest neighbor algorithm than the case that all features are used. To achieve this goal, a novel filter evaluation criterion (named MIRFFS) based on the concepts of mutual relevance, RelifF [27] and Fisher Score [28] is proposed, and using this proposed criterion, the standard DE and multi-objective DE (MODE) based feature selection approaches are developed. Furthermore, a widely used existing filter based criterion (MIFS) is also redesigned as fitness function for single objective and multi-objective DE to develop filter based approaches. These four developed feature selection approaches will be examined and evaluated on benchmark problems of varying difficulty. Specifically, we will investigate

- the performance of the four algorithms (i.e. single objective and multi-objective DE approaches based on MIRFFS and MIFS) on reducing the number of features and improving the classification performance over using all features,
- the performance of the single objective DE approach based on MIRFFS versus based on MIFS,
- the performance of the multi-objective DE approach based on MIRFFS versus based on MIFS,
- the performance of the multi-objective DE approaches versus the single-objective DE approaches, and
- the performance of all DE filter approaches versus traditional approaches.

1.2. The organization of the paper

The rest of the paper is organized as follows. Section 2 gives an outline of the basic DE algorithm and provides a background on information theory, feature ranking and recent studies related to feature selection, especially filters. Section 3 describes the DE based feature selection approaches using the proposed and existing criteria. Section 4 shows the experimental design and Section 5 presents the experimental results with discussions. Finally, Section 6 concludes the paper and provides an insight into the future trends.

2. Background

This section provides a background concerning the differential evolution, multi-objective optimization, information theory and recent filter approaches.

2.1. Differential evolution

Differential evolution (DE) is a search algorithm proposed by Storn and Price [29] in 1997. DE belongs to the class of evolutionary algorithms in EC techniques that applies biologically inspired operators such as crossover, mutation and selection. The algorithm uses mutation to search in the solution space and applies selection to direct search toward the prospective regions in the solution space. Furthermore, non-uniform crossover plays a critical role in the algorithm performance, where one parent influenced the child more than others. The crossover operator constructs trial vectors by efficiently shuffling useful information in the population and recombine them to find better solutions [29]. In DE, solution vectors are first randomly initialized. These solutions are then improved by applying the three operators: mutation, crossover and selection. In DE, greedy selection is applied between each generated solution and a mutant solution to update the population. The basic steps of DE are summarized below:

1) Initialization. DE first randomly produces solution vectors in the search space. Each solution vector defined as $X_i = \{x_{i1}, x_{i2}, x_{i3}, ..., x_{ij}, ..., x_{iD}\}$ is generated by:

$$x_{ij} = x_j^{min} + U(0,1)(x_j^{max} - x_j^{min}) \tag{1}$$

where $i = \{1, 2, ..., NP\}$ and NP is population size; $j = \{1, 2, ..., D\}$; D is the dimensionality of the search space; $U(0, 1)$ is the random variable uniformly distributed between (0,1); x_j^{min} and x_j^{max} are predefined minimum and maximum values of parameter j.

2) Mutation. Each solution vector undergoes mutation to expand the search space. A mutant solution $\hat{X}_i$ is generated by:

$$\hat{X}_i = X_{r1} + F(X_{r3} - X_{r2}) \tag{2}$$

where F is the scaling factor predefined within the range of [0,1] and X_{r1}, X_{r2} and X_{r3} are randomly chosen solution vectors which must satisfy

$$r1 \neq r2 \neq r3 \neq i \tag{3}$$

where i is the current solution vector. Eq. (3) indicates that NP must be chosen at least 4.

3) Crossover. The non-uniform crossover is applied between the mutant and parent solution vectors by:

$$u_{id} = \begin{cases} \hat{x}_{id}, & \text{if } rand(d) \leq CR \text{ or } j = rn_i, \\ x_{id}, & \text{otherwise,} \end{cases} \tag{4}$$

where CR is the user predefined crossover rate, $rand(d)$ is the uniformly generated number between [0,1] for parameter j, rn_i is the randomly chosen index and u_{id} is the dth parameter of a trial vector $U_i = \{u_{i1}, u_{i2}, ..., u_{ij}, ..., u_{iD}\}$.

4) Selection. Greedy selection is applied between the current solution X_i and trial solution U_i. If U_i is better than X_i, U_i is represented in next generations instead of X_i.

The population is updated by applying mutation, crossover and selection operators from generation to generation until a stopping criterion is met.

2.2. Multi-objective optimization

Many problems involve two or more objectives that are conflicting to each other. Multi-objective optimization is concerned with more than one objective function to be optimized simultaneously. This type of problems have more than one optimal solutions, typically referred as Pareto-optimal solutions.

Let $f(x) = (f_1(x), f_2(x), ..., f_{n_o}(x)) \in O \subseteq \mathbb{R}^{n_0}$ be an objective vector comprising of multiple (n_0) conflicting functions and let $S_f \subseteq S$ (where S is the search space) represents the feasible space constrained by n_g inequalities and n_h equality constraints;

$$S_f = \{x : g_m(x) \leq 0, h_l(x) = 0, m = 1, ..., n_g; l = 1, ..., n_h\} \tag{5}$$

where $g_m(x)$ and $h_l(x)$ are constraints. Using this notation, a multi-objective (minimization) problem can be formulated as follows:

$$\text{minimize } f(x) \text{ subject to } x \in S_f \tag{6}$$

When there are multiple objectives, for two solutions y and z, y dominates z if y is not worse than z in all objective functions and better than z in at least one objective function:

$$\forall k : f_k(y) \leq f_k(z) \wedge \exists k : f_k(y) < f_k(z) \tag{7}$$

A solution $x^* \in S_f$ is defined as a Pareto-optimal (non-dominated) solution if there does not exist a solution $x \neq x^* \in S_f$ that dominates x^*. The set of all non-dominated solutions form a Pareto-optimal front surface, known as Pareto front.

2.3. Information theory

Information theory was first proposed for communication theory to find limits concerning data compression and transmission rate [30]. Due to its suitability, now it has been used in a variety of fields, including natural language processing, cryptography, pattern recognition and data analysis [31]. The basic concepts of information theory are as follows.

1) Entropy (*H*). Entropy is a measure of uncertainty of a random variable. The uncertainty is related to the probability of occurrence of an event, defined by Eq. (8). While high entropy means that each value of the variable is about the same probability of occurrence, low entropy means that each value of the variable is about the different probability of occurrence.

$$H(X) = -\sum_k p(x_k) \log_2 p(x_k) \tag{8}$$

where X is a random variable and $p(x_k) = Pr\{X = (x_k), x_k \in X\}$ is the mass probability. The joint and conditional entropy of two random variables X and Y are defined as follows:

$$H(X, Y) = -\sum_{k,z} p(x_k, y_z) \log_2 p(x_k, y_z) \tag{9}$$

$$H(X|Y) = -\sum_{k,z} p(x_k, y_z) \log_2 p(x_k|y_z) \tag{10}$$

where $X = \{x_1, x_2, ...x_k, ..., x_n\}$ and $Y = \{y_1, y_2, ...y_z, ..., y_m\}$.

2) Mutual information. The mutual information is a measure of mutual dependence between random variables. It therefore provides a way to evaluate the relevance of a feature subset. Mutual information between any two variables X and Y can be expressed as follows:

$$I(X; Y) = -\sum_{k,z} p(x_k, y_z) \log_2 p\left(\frac{p(x_k, y_z)}{p(x_k).p(y_z)}\right) \tag{11}$$

Eq. (11) can be also rewritten as $I(X; Y) = H(X) + H(Y) - H(X, Y)$ or $I(X; Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$.

2.4. Recent studies on filter approaches

For a given data $X \in \mathbb{R}^{N \times M}$ and the class labels $Y \in \mathbb{R}^{N \times 1}$ where N is the number of instances (samples) and M is the number of features, the aim of a filter-based feature selection approach is to choose a feature subset with size m based on some prior knowledge or statistical criterion, where $m < M$. The optimal feature subset provides the maximum combined information content of all selected features with respect to the class labels. However, it is an NP-hard combinatorial problem and the optimal feature subset can only be obtained by a brute-force (exhaustive) search [1]. Due to its difficulty and complexity, there has been extensive research on filter approaches. We consider these approaches in three subsections.

1) Traditional Filter Approaches. One of the simplest filter approaches is to rank the features with target to the class labels based on a suitable criterion or metric. Pearson correlation coefficient [32] ranks features in descending order with target to the class labels using the mean and standard deviations. Then, a predictor is applied on M nested subsets and the subset with the lowest validation error is chosen. Although it is simple to implement and computationally efficient, it assumes all features are independent and is only able to detect linear relationship between each feature and the class labels. Another simple filter approach is Laplacian Score [33] which does not only consider features with larger variances, but also considers the features with stronger locality preserving ability. After ranking features according to the Laplacian values, it uses the K-means clustering method to choose the best k features. Laplacian Score has been proved effective and efficient. However, the shortcomings of K-means also lead problems in Laplacian Score. Some improved versions of Laplacian Score can be found in [34,35]. In contrast to Pearson Correlation and Laplacian Score, Fisher Score [28] is a supervised ranking approach and it orders features according to their discriminant ability. It evaluates features individually; thus, it cannot consider redundancy (no correlations amongst filters). Relief and its extended version (ReliefF) [27] assign a ranking score for each feature individually based on a k nearest neighbor algorithm. Although it is one of the best representative samples for filter approaches, it does not unfortunately consider redundancy which is the for other mentioned traditional filter approaches. Hall [36] developed correlation based feature selection (CFS) as a heuristic method for feature selection, which aims to find a feature subset highly correlated to the class label and uncorrelated with each other. Systematical uncertainty was used in [36] to evaluate the correlation, but it cannot handle relationship among multiple variables.

2) Information Theoretic Filter Approaches. Since traditional filter approaches rely solely on the relationship between features and the class labels (referred as 'relevance'), they cannot work well in the presence of dependent features (e.g. overlapping information amongst the features). By considering the information between features (referred as 'redundancy'), information theoretic filter approaches can be treated as an alternative to the traditional approaches. One of the most famous approaches is mutual information feature selection (MIFS) [13]. MIFS is a greedy heuristic approach consisting of following steps: 1) add the highest relevant feature to the empty subset S and 2) add next $(m-1)$ features to the subset S sequentially based on criterion, defined by Eq. (12);

$$MIFS = \max_{i \in Q-S}\left(\underbrace{I(x_i; y)}_{relevance} - \underbrace{\beta \sum_{s \in S} I(x_i; x_s)}_{redundancy}\right) \tag{12}$$

where Q is the initial feature set, x_i is the *ith* feature in Q which is not selected for subset S yet, x_s is the selected feature in the subset S, y is the class labels and β is the predefined parameter satisfying balance between relevance and redundancy.

As seen in Eq. (12), MIFS requires a user-specified parameter (β) that may vary according to the size of feature subset, but it is hard to determine. To avoid the fine tuning of the specified parameter, Peng et al. [14] improved the MIFS criterion by introducing the maximum relevance and minimum redundancy method (mRmR), defined in Eq. (13);

$$mRmR = \max_{i \in Q-S}\left(I(x_i; y) - \frac{1}{|S|}\sum_{s \in S} I(x_i; x_s)\right) \tag{13}$$

where $|S|$ is the size of subset S.

mRmR follows the same methodology as in MIFS, but performs better than MIFS. Estevez et al. [37] normalized the relevance component (between two features) of mRmR by dividing with the minimum entropy of the two features. Brown [38] added the class-conditional correlations to Eq. (12), referred as the first-order utility (FOU). Al-Ani and Deriche [39] introduced a criterion, named as mutual information feature selection (MIEF). MIEF achieved better results than MIFS in image sets. Zhang et al. [40] proposed a two-stage feature selection approach for text classification, which ranks features based on gain ratio and then try to select best feature subset among high ranked features based on the classification performance obtained by a classifier. Freeman et al. [41] presented a comprehensive comparative study of recent filter approaches, including ReliefF, mRmR, CMIM and FOU. Yu et al. [42] developed a comprehensive library for feature selection which introduces measures, such as Fisher Score and mutual information to calculate correlations between features. Due to the challenges of two-way relationships in high dimensional problems, Chen et al. [43] developed a new feature selection approach using high order inter-correlation (redundancy). To verify the effectiveness of the proposed approach, a comprehensive comparative study was made by comparing it with seven representative feature selection methods, including mRmR, ReliefF and CMIM. However, the computational cost may be extremely increased proportional to the number of features due to more than two relations between features. In addition, mutual information has also been used for feature selection in multi-label classification problems [44] and intrusion detection systems [45]. Due to the difficulties on calculating probabilities of continues variables via standard mutual information, fuzzy mutual information measures have also been proposed for solving feature selection tasks [46,47].

3) EC based Filter Approaches. As information theory and traditional feature selection approaches are mostly greedy heuristic approaches, they often cannot search the possible feature space effectively. Therefore, their performance may deteriorate in large-scale datasets. Therefore, researchers have applied EC techniques to feature selection. Ge and Hu [18] proposed a feature selection approach that combines GA and mutual information (FSGM). In FSGM, FOU was chosen as the objective function. The results show that FSGM was superior to sequential forward selection and ReliefF. However, it was not compared with GA based on other existing mutual information criteria like MIFS and mRmR. Huang and Rong [48] introduced a two stage (filter-wrapper) GA to increase the classification accuracy. While the filter stage as an inner loop tries to optimize the improved MIFS criterion with the parameter free conditional mutual information, the wrapper stage as an outer loop tries to optimize the kappa statistic. Cervante et al. [17] introduced a binary PSO based information theoretic feature selection approach by adopting mRmR as an objective function. However, the parameter in the objective function that compromises between the relevance and redundancy needs to be predefined by a user. Nguyen et al. [49] integrated mRmR criterion as a local search into wrapper based PSO, and they [50] further investigated the use of mutual information estimation in PSO for feature selection to be applied on continuous datasets. In [50], mRmR is redesigned as the objective function in a PSO framework using pairwise mutual information instead of multivariate mutual information due to its computational efficiency. Al-Ani [51] proposed an ACO based filter approach (ANT) based on MIEF for the classification of speech segments. According to the results, it was superior to GA. Khushaba et al. [52] extended the ANT filter approach by hybridizing with DE (referred as ANTDE). It was seen that the results obtained by ANTDE were very promising when compared to BPSO, GA and ANT. Moradi and Rostami [53] introduced a two-stage ACO based filter approach based on graph representation and a community detection algorithm. The results indicated that the introduced approach was superior to a number of filter approaches such as mRmR and ReliefF and Fisher Score. However, it may be computationally intensive due to the representation

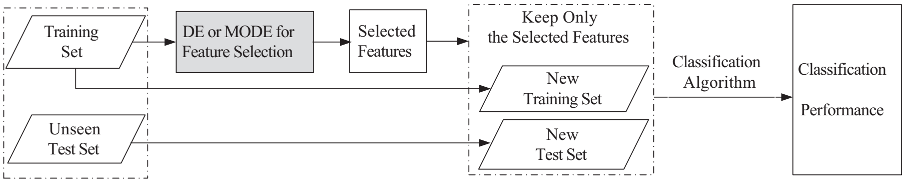

Fig. 1. Overall structure.

scheme. Xue et al. [54] considered mRmR criterion as a multi-objective problem through non-dominated sorted GA (NSGAII) and strength Pareto evolutionary algorithm 2 (SPEA2). It was observed that multi-objective schemes can provide more promising results than single-objective schemes. Rough set theory and multivariate mutual information are used in a GA based two-objective framework for feature selection[55], but both rough set theory and multivariate mutual information are expensive measures.

Although a number of filter approaches have been proposed in the literature, there are still some open issues that need to be considered. First, there are just only few DE based filter approaches, especially inspired by information theory, although DE is one of the most robust and stable EC techniques and has been successfully applied to a variety of applications [5]. Second, most of the existing filter based feature selection approaches are single-objective and the idea of simulating feature selection as a multi-objective problem has just come into consideration in recent years. Third, most of the recent information theoretic feature selection criteria have been developed by introducing similar variants of MIFS. In other words, only a few new approaches have been proposed as an alternative to MIFS and mRmR in the literature.

3. Proposed filter based approaches

3.1. The overall structure

The overall structure is shown in Fig. 1, where the training set is fed to the proposed DE or multi-objective DE (MODE) based feature selection algorithms to select a small number of informative features. Then, the features that are not selected will be removed from both the training set and the test set. Finally, a classification algorithm is applied to the new training and test sets to evaluate the classification performance. This system is designed to avoid feature selection bias (which is a common issue in a large number of papers [56]), and keep the test set completely unseen from the feature selection algorithms. The performance of the DE or MODE based feature selection methods will be evaluated based on the achieved classification accuracy and the number of selected features.

The rest of this section describes the proposed four feature selection algorithms, particularly the new fitness functions, which are the key in any feature selection algorithm. Section 3.2 describes the two algorithms, the single objective algorithm (DE$_{mi}$) and the multi-objective algorithm (MODE$_{mi}$), which are based on the most well-known information theoretic feature selection criterion (i.e. MIFS) with modifications. Section 3.3 describes the two algorithms, the single objective algorithm (DE$_{mirf}$) and the multi-objective algorithm (MODE$_{mirf}$), which are based on our newly develop evaluation criterion (referred as MIRFFS) inspired by Mutual Information, ReliefF and Fisher Score. Four algorithms (instead of a single algorithm) are developed as a systematic research to investigate the performance of DE, information theory and feature ranking for feature selection.

3.2. DE for feature selection based on MIFS

DE based on MIFS (DE$_{mi}$): As mentioned in Section 2, MIFS is a well-known representative information theoretic approach. However, MIFS considers features individually and applies a greedy approach to form the feature subset, i.e, it does not search the solution space effectively. Therefore, DE is chosen and Eq. (12) is reformulated into Eq. (14) to be used as the fitness function in DE to guide the search to find optimal feature subsets. Note that normalization is implemented for the calculations of mutual information values to keep the consistency between possible feature subsets.

$$fit_{mi}(S) = \max\left(\underbrace{\sum_{x_k\in S} NI(x_k;y)}_{relevance} - \beta\underbrace{\sum_{x_k\in S}\sum_{x_l\in S} NI(x_k;x_l)}_{redundancy}\right) \tag{14}$$

where $k\neq l$, S is the selected feature subset, β is the predefined value, x_k and x_l are the kth and lth selected features, and y is the class label. $NI(x_k;\ y)$ is the *normalized* $I(x_k;\ y)$ representing mutual relevance, and $NI(x_k;\ x_l)$ is the *normalized* $I(x_k;\ x_l)$ representing mutual redundancy:

$$NI(x_k;y) = \frac{I(x_k;y)}{\sqrt{\sum_{m=1}^{M} I(x_m;y)^2}} \tag{15}$$

$$NI(x_k;x_l) = \frac{I(x_k;x_l)}{\sqrt{\sum_{m=1}^{M-1}\sum_{j=m+1}^{M} I(x_m;x_j)^2}} \tag{16}$$

where M is the total number of features in the dataset.

A new method named DE$_{mi}$ is proposed by using DE as the search method with Eq. (14) as the fitness function to find optimal feature subsets. The representation of an individual is a M-bit continuous vector representing a possible feature subset where the possible values in the vector is in the range of [0, 1]. If any dimension of an individual is greater than 0.5, the corresponding feature is selected; otherwise, it is not selected. The pseudo-code of the DE based on MIFS can be found in Algorithm 1 . If any value (or gene) in the mutant individual is out of the range [0,1], that value is constrained within the range by Eq. (17), which is the most common way to deal such with out-of-range cases.

$$\begin{cases} U_{ij}(t) = 0, & \text{if } \forall j \in \{1, \dots, M\} : U_{ij}(t) < 0, \\ U_{ij}(t) = 1, & \text{if } \forall j \in \{1, \dots, M\} : U_{ij}(t) > 1, \end{cases} \tag{17}$$

MODE based on MIFS (MODE$_{mi}$): Eq. (14) considers both the relevance between features and the class labels, and the redundancy among features in a weighted manner, i.e., β that provides the balance between these two components needs to be predefined. In most cases, users may tend to make an informed decision from available feature subsets. Therefore, it is necessary to consider the two components in Eq. (14) in a multi-objective design with the objectives of maximizing the relevance and minimizing the redundancy.

DE was first proposed as a single objective optimizer for continuous problems. To apply DE to multi-objective problems, a new selection mechanism (see Section 2.1) should be reformed according to more than one objective. Although there exist various multi-objective DE variants in the literature [26], multi-objective DE

Algorithm 1: Pseudo-code of DE_{mirf} (and DE_{mi})

```
begin
    Calculate mutual relevance between features in both $\text{DE}_{mirf}$ and $\text{DE}_{mi}$;
    Calculate order values of all features using ReliefF and Fisher Ranking in $\text{DE}_{mirf}$;
    Initialize individuals using Eq.(1);
    Evaluate the fitness of individuals using Eq.(19) for $\text{DE}_{mirf}$ (Eq.(14) for $\text{DE}_{mi}$);
    for $iter \leftarrow 1$ to $MaxIter$ do
        foreach individual $i$ do
            Select three individuals $r_1$, $r_2$ and $r_3$ randomly;
            Generate a mutant solution $\hat{X}_i$ by applying the mutation operator shown by Eq.(2);
            Generate a trial vector $U_i$ by applying the crossover operator shown by Eq.(4);
            Evaluate fitness value of the trial vector $U_i$ using Eq.(19) for $\text{DE}_{mirf}$ (and Eq.(14) for $\text{DE}_{mi}$);
            // Greedy selection:
            if fitness of $U_i$ is better than $i$ then
                Use $U_i$ to replace $i$;
            else
                discard $U_i$;
    Collect the features selected by the individual with the best fitness value;
    Calculate the classification accuracy of the selected features on the test set;
    Return the individual and its classification accuracy rate;
```

Algorithm 2: Pseudo-code of MODE_{mirf} (and MODE_{mi})

```
begin
    Calculate mutual relevance between features in both $\text{MODE}_{mirf}$ and $\text{MODE}_{mi}$;
    Calculate order values of all features using ReliefF and Fisher Ranking in $\text{DE}_{mirf}$;
    Initialize individuals by Eq.(1);
    Evaluate the objective values of each individual;
    // Three objectives shown as relevance, ReliefF ranking and Fisher Ranking in Eq.(19) for $\text{DE}_{mirf}$
    // Two objectives shown as relevance and redundancy in Eq.(14) for $\text{MODE}_{mi}$
    for $iter \leftarrow 1$ to $MaxIter$ do
        foreach individual $i$ do
            Select three individuals $r_1$, $r_2$ and $r_3$ randomly;
            Generate a mutant solution $\hat{X}_i$ using the mutation operator, Eq.(2);
            Generate a trial vector $U_i$ using the crossover operator, Eq.(4);
            Evaluate the objectives of trial vector $U_i$;
            // Pareto-dominance-based selection:
            if $i$ does not dominate $U_i$ then
                Use $U_i$ to replace $i$;
            else
                discard $U_i$;
    Find the Pareto non-dominated solutions (feature subsets) in the final generation of the population ;
    Calculate the classification accuracy of the feature subsets on the test set;
    Return the feature subsets and their testing classification accuracy rates;
```

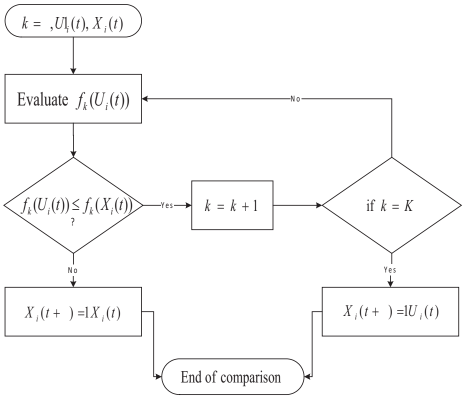

Fig. 2. The flowchart of dominance-based selection.

(MODE) [57,58] is chosen as a multi-objective DE optimizer due to its simplicity and low time complexity. It is easy to implement and does not include any complex structure such as non-dominated sorting and archive keeper. MODE uses dominance-based selection inspired by Lampinen's criterion [59] to determine Pareto-optimal solutions. The dominance-based selection is defined by Eq. (18) and its general implementation is presented in Fig. 2. The pseudo-code of using MODE for multi-objective feature selection, i.e. the proposed MODE_{mi} algorithm, is shown in Algorithm 2 . The possible feature subset representation scheme of an individual in MODE_{mi} is same as DE_{mi} within the range of [0, 1]. If any position of an evolved is out of the range, that position is constrained within the range by Eq. (17).

$$X_i(t+1) = \begin{cases} U_i(t), & \text{if } \forall k \in \{1, \ldots, K\} : f_k(U_i(t)) \leq f_k(X_i(t)), \\ X_i, & \text{otherwise.} \end{cases} \tag{18}$$

where t is the cycle number and K is the total number of objectives.

3.3. DE for feature selection based on the new criterion (MIRFFS)

DE based on MIRFFS (DE_{mirf}): Although MIFS is a well-known information theoretic feature selection approach, more than two-way relationships between features are mostly ignored or under-estimated by MIFS and its variants, i.e., they generally focus on the relationships between pair of features as shown in Eq. (12)–(14). Accordingly, it is not possible to fully evaluate the mutual redundancy among features. To address the problem, high order interactions can be evaluated via conditional mutual information or other mutual information techniques. However, the computation of high order interactions is highly computationally expensive and substantially increases algorithmic complexity. In order to reduce the time complexity and find better feature subsets, it is necessary to propose a new criterion.

In this study, we propose a new filter criterion inspired by feature ranking and information theory, in particular mutual information, ReliefF and Fisher Score, so the new criterion is named MIRFFS and defined by Eq. (19). In contrast to MIFS and its variants, MIRFFS aims to eliminate low ranked features detected by

ReliefF and Fisher Score.

$$Fit_{mirf}(S) = \max\left(\sum_{x_k\in S}\underbrace{NI(x_k;y)}_{relevance} - \beta\left(\sum_{x_k\in S}\underbrace{NRelief_{order}(x_k)+NFisher_{order}(x_k)}_{ranking}\right)\right) \quad (19)$$

where $NI(x_k, y)$ is the normalized mutual relevance between *kth* feature and the class labels, defined by Eq. (15);

NRelief: $NRelief_{order}(x_k)$ is the normalized ranking/order values of *kth* feature, determined by Eq. (20);

$$NRelief_{order}(x_k) = \frac{Relief_{order}(x_k)}{p * \sum_{m=1}^{M} Relief_{order}(x_k)^2} \quad (20)$$

where $Relief_{order}(x_k)$ is the order value of *kth* feature between [1, *M*], where 1 means *k*th feature is ranked as top 1 (the best) and *M* means the worst. *p* is a parameter to control the balance in different datasets. The *Relief* score for feature x_k can be calculated by Eq. (21) (details can be seen in [27]), where *P* means probability:

$$ReliefF(x_k) = P(x_k\text{value}|\text{different class}) - P(x_k\text{value}|\text{same class}) \quad (21)$$

NFisher: $NFisher_{order}(x_k)$ is the normalized order value of *k*th feature, determined by Eq. (22);

$$NFisher_{order}(x_k) = \frac{Fisher_{order}(x_k)}{p * \sum_{m=1}^{M} Fisher_{order}(x_m)^2} \quad (22)$$

where *M* is the total number of features in the dataset; $Fisher_{order}(x_k)$ is the order value of *kth* feature between [1, *M*] among *M* features according to Fisher Score values, calculated by Eq. (23) to maximizing the between-class scatter and minimizing the within-class scatter (details can be seen in [28]);

$$FisherScore(x_k) = \sum_{n=1}^{N}\left|\frac{\mu_i^k - \mu_j^k}{{\sigma_i^k}^2 - {\sigma_j^k}^2}\right| \quad (23)$$

where μ_i^k and μ_j^k are the mean of the *k*th feature in the *i*th and *j*th classes, and σ_i^k and σ_j^k are the corresponding standard deviation values.

As seen in Eqs. (20) and (22), the normalized order values are decreased inversely proportional to *M* value, but as seen in Eq. (15), the mutual relevance values will be increased proportional to *M* value. It is therefore difficult to keep balance between relevance and ranking for high dimensional datasets. With *p* parameter, it is aimed to keep normalized order values at a reasonable level for high dimensional problems. *p* parameter is set to 1/2, 1/3 or 1/4 for the datasets including more than 50 features; otherwise, it is chosen as 1.

By using Eq. (19) as the fitness function in DE, a new feature selection approach is proposed in this study, which is named DE_{mirf}. The individual representation scheme of this approach DE_{mirf} is same as DE_{mi}. The pseudo-code of DE_{mirf} can be illustrated in Algorithm 1, where the major difference between DE_{mi} and DE_{mirf} is the fitness function.

An example: We include the following example to show the calculation of fitness function in DE_{mirf}. Let, $Z = \{x_1, x_2, ..., x_8\}$ be a dataset comprising of 8 features, β is set to 1. After calculations, features (from x_1 to x_8) are ordered as {7, 4, 8, 3, 2, 6, 1, 5} in terms of ReliefF, and are ordered as {7, 8, 6, 2, 3, 5, 1, 4} in terms of Fisher Score. The normalized mutual relevance, Fisher Score ranking and ReliefF ranking values are calculated according to Eqs. (15), (22) and (20), which are shown as follows:

$$NI = \{0.3472, 0.0130, 0.0689, 0.2223, 0.3591, 0.1534, 0.7863, 0.2330\}$$

$$NRelief_{order} = \{0.4901, 0.2801, 0.5601, 0.2100, 0.1400, 0.4201, 0.0700, 0.3501\}$$

$$NFisher_{order} = \{0.4901, 0.5601, 0.4201, 0.1400, 0.2100, 0.3501, 0.0700, 0.2801\}$$

Let [0.30, 0.80, 0.65, 0.23, 0.75, 0.45, 0.15, 0.85] be an individual in DE. The features ({x_2, x_3, x_5, x_8}) who's the corresponding positions in the individual are greater than 0.5 are selected. According to the selected feature subset, the fitness value is computed as -2.1268 via Eq. (19). Note that the mutual relevance, ReliefF and Fisher Score values of all available features are computed only once before the evolutionary process of DE. During evolutionary process of DE, these values can be used to calculate the fitness value of each individual.

MODE based on MIRFFS ($MODE_{mirf}$): As in MIFS, MIRFFS (Eq. (19)) uses β parameter to provide the balance between the mutual relevance and the feature ranking. Furthermore, *p* parameter is used in normalization process of ranking values to keep the ranking values at a reasonable range for high dimensional problems. The determination of optimal parameter values is generally time consuming and the performance of DE_{mirf} highly dependents on these parameters. Therefore, MIRFFS needs to be considered in multi-objective DE design. In contrast to $MODE_{mi}$ and existing multi-objective studies in the literature, $MODE_{mirf}$ is proposed in this work to optimize three objectives, which are mutual relevance, ReliefF ranking and Fisher Score ranking. By simultaneously optimising these three objectives, the archived feature subsets are expected to achieve better classification performance by automatically finding a balance among these criteria. The representation scheme of $MODE_{mirf}$ is same as in $MODE_{mi}$ and the pseudo-code of $MODE_{mirf}$ can be illustrated in Algorithm 2.

4. Experimental design

To examine the performance of the feature selection approaches, ten datasets from UCI machine learning repository [60], one biomedical data (DNA) and one text classification data (listed in Table 1) are chosen, including different numbers of features, samples and classes. Since mutual information cannot be computed on continuous data, all chosen datasets are categorical data. For each dataset, 70% of the samples are randomly selected as the training set and the remaining (30%) samples are as the test set. Notice that we also consider the distribution of instances over classes during the data division process. To cope with missing values in some datasets, there exist a number of techniques in the

Table 1
Datasets.

Dateset	Number of features	Number of classes	Number of examples
Lymph	18	4	148
Spect	22	2	267
Leddisplay	24	10	1000
Soybean large	35	19	307
Connect	42	3	3196
Promoter	57	2	106
Splice	60	3	3190
Optic	64	10	5620
Audiology	68	24	226
Coil2000	85	2	9000
DNA	180	2	3186
PCMAC	3289	2	1943

literature such as imputation, recovering and deletion. As only three datasets used for comparisons contain a small number of missing values, we eliminate data instances for datasets which include any missing value.

The filter approaches are first run on the training set to get the optimal feature subset(s). Then, the performance of the optimal feature subset(s) is evaluated by the learning/classification algorithm on the test set. Note that the learning algorithm is solely applied to the test set to obtain the classification performance of the optimal feature subset(s). Due to its simplicity and popularity, the learning algorithm is selected as K-nearest neighbor (KNN), where K is set to 5 as in [54] in the experiments.

The β values in Eq. (19) are set to 0.9, 0.7, 0.5, 0.3 and 0.1, respectively. For the comparative study of multi-objective and single objective approaches, the β value of single-objective approaches is set to 0.3 that generally provides the best classification performance.

The experiments are conducted for 30 runs. In the experiments, the population size is set to 50 and the maximum number of generations is defined as 50 for all approaches. For single objective approaches, the scaling factor and the crossover rate are experimentally chosen as 0.8 and 0.7, and for multi-objective approaches, the scaling factor and the crossover rate are set to 0.5 and 0.2, respectively as suggested in [20]. To show the significant difference between the proposed and existing criteria, the Wilcoxon Rank Sum test is performed with the significance level of 0.05. If the p-value is equal or smaller than 0.05, the approach based on the proposed criterion performs significantly better than based on the existing criterion at 95% of confidence.

Two traditional correlation based filter approaches (CfsF and CfsB [61]) and one wrapper approach (GSBS [62]) are employed for comparisons in the experiments. While CfsF performs forward search, CfsB and GSBS performs backward search. The experiments of the three traditional approaches are performed in Waikato Environment for Knowledge Analysis (WEKA) [63] platform. To make fair comparisons, the experiments of all approaches are first conducted using the same 10-fold cross-validation on the same training set to obtain feature subsets. Then, the same classifier is used to evaluate the classification performance of the feature subsets obtained by the approaches on the same test set.

5. Results and discussions

In this section, results are mainly considered in two subsections. First, we analyze the classification performance and the number of features obtained by the approaches: DE_{mirf} vs. DE_{mi}, $MODE_{mirf}$ vs. $MODE_{mi}$, multi-objective vs. single objective and comparisons with traditional approaches. Second, we compare the computational time of the approaches: DE_{mirf} vs. DE_{mi}, and $MODE_{mirf}$ vs. $MODE_{mi}$.

5.1. Comparisons between DE_{mirf} and DE_{mi}

Table 2 shows the results of DE_{mirf} and DE_{mi} with β values from 1 to 0.1 in descending order. In Table 2, in the first column, below the caption of each dataset, the numbers correspond to the number of available features and the classification accuracy using all features. The standard deviation values of classification accuracy are presented in brackets and the mean values of feature subset size appear below the results of the classification accuracy for each approach over 30 independent runs. The results of the Wilcoxon Rank Sum Test are shown via ‘Sig. Test’, where ‘+’ or ‘−’ means the classification performance of DE_{mirf} is significantly better or worse than DE_{mi} and ‘=’ means there is no significant difference between DE_{mirf} and DE_{mi}.

According to Table 2, it can be observed that with at least two of the β values, DE_{mi} can generally evolve a small number of features and achieve similar or better classification performance than using all features except for the Promoter, Optic and Coil2000 datasets. Although DE_{mi} performs slightly worse than using all features in terms of the classification accuracy in the Coil2000 dataset, it can select only around 30 features from the available 85 features. In the Optic dataset, it can select only around 48 features from the available 64 features and achieve 98.75% classification accuracy (which is very close to 98.87% obtained using all available features). Therefore, it can be suggested that DE_{mi} has the potential to reduce the feature subset and increase the classification accuracy.

According to Table 2, it can be also observed that DE_{mirf} evolve a small number of features and achieve similar or better classification performance than using all features for all values of β in the Spect, Leddisplay, Connect (except for 1 case of β), Splice, Audiology, DNA (except for 1 case of β) and PCMAC datasets. In the Lymph and Soybean datasets, it can perform better than using all features in 3 values of β. Only, it cannot obtain better performance in the Promoter and Optic datasets, but the classification performance of DE_{mirf} is very close to the results obtained by using all features. Therefore, DE_{mirf} can significantly reduce the dimensionality of the data and maintain or increase the classification performance. Further, it reaches this success with more β options than DE_{mi}.

As seen in Table 2, the classification performance and the number of features tend to increase inversely proportional to the β value in both DE_{mirf} and DE_{mi}. It is also seen that both approaches mostly achieve the best performance when the β value is 0.3. Comparing DE_{mirf} with DE_{mi}, the average size of the feature subsets evolved by DE_{mirf} is smaller than DE_{mi} in most cases. Not only obtaining smaller feature subset size, but also DE_{mirf} provides higher classification performance in most cases. Further, the classification performance of DE_{mirf} is significantly better than DE_{mi} in almost all cases except for the Lymph, Promoter, Optic and DNA datasets. Although there is generally no difference between DE_{mirf} and DE_{mi} in the Promoter and Optic datasets, DE_{mirf} selects a smaller number of features. Thus, DE_{mirf} can be also treated as successful in the Promoter and Optic datasets. It can be suggested that the proposed criterion outperforms the most-widely used existing criterion in terms of the classification accuracy and the number of features.

Generally, DE_{mirf} and DE_{mi} can be applied to feature selection problems. DE_{mirf} which is the combination of feature ranking and mutual information is a better feature selection approach than DE_{mi}. However, it is unclear whether more features can be removed and the classification accuracy can still be maintained or even increased. Furthermore, the parameter to balance between the components in both the MIRFFS and MIFS criteria is difficult to predefine in advance. Therefore, it would be interesting to consider feature selection as a multi-objective problem to explicitly examine the trade off between the classification accuracy and number of features.

5.2. Comparisons between $MODE_{\mathrm{mirf}}$ and $MODE_{\mathrm{mi}}$

In the experiments, single objective approaches obtain a single feature subset/solution in each independent run (30 feature subsets for the 30 independent runs). Multi-objective approaches obtain a set of nondominated solutions in each independent run. In order to compare the single objective algorithms with the multi-objective algorithms, the 30 sets of solutions obtained by multi-objective approaches are collected into a union set. In the union set, the classification performance of the solutions that have the same subset size are averaged. A new set of average solutions is referred as the “average” front. In addition to the “average” front,

Table 2
The results of single objective DE based on MIFS and MIRFFS over KNN.

Dataset	Method	β=1	β=0.9	β=0.7	β=0.5	β=0.3	β=0.1
	DE$_{mi}$	84.60 (2.14)	83.49 (0.60)	85.71 (4e-16)	80.95 (1e-16)	88.09 (3e-16)	88.09 (3e-16)
Lymph		5.26	5	6.03	8	11	17
(18,88.09%)	**DE$_{mirf}$**	83.33 (5e-16)	83.33 (5e-16)	88.09 (3e-16)	80.95 (1e-16)	88.09 (3e-16)	88.09 (3e-16)
		5	5	8	9	12	17
	Sig. Test	−	=	+	=	=	=
	DE$_{mi}$	80.00 (3e-16)	80.00 (3e-16)	80.00 (3e-16)	81.25 (0)	78.75 (2e-16)	77.50 (2e-16)
Spect		8	8.03	10	12	14	20
(22,78.75%)	**DE$_{mirf}$**	81.25 (0)	81.25 (0)	81.25 (0)	81.25 (0)	80.00 (3e-16)	78.75 (2e-16)
		2	3	7	9	14	22
	Sig. Test	+	+	+	=	+	+
	DE$_{mi}$	83.23 (2.27)	73.61 (1.85)	93.26 (1.52)	93 (2e-16)	88.88 (0.42)	90.00 (4e-16)
Leddisplay		7.3	8.9	9.1	12	16	24
(24,90.00%)	**DE$_{mirf}$**	100 (0)	100 (0)	100 (0)	100 (0)	100 (0)	100 (0)
		7	7	7	7	7	7
	Sig. Test	+	+	+	+	+	+
	DE$_{mi}$	72.67 (7.65)	76.62 (5.86)	78.37 (4.05)	83.11 (2.42)	84.29 (1.98)	85.70 (0.96)
Soybean		10.03	10.63	12.4	16.63	22.4	33.76
(35,85.53%)	**DE$_{mirf}$**	82.98 (3.60)	82.06 (2.79)	83.15 (1.39)	85.87 (1.28)	85.92 (1.04)	83.51 (0.89)
		10.6	11.23	13.3	16.53	21.23	27.76
	Sig. Test	+	+	+	+	+	−
	DE$_{mi}$	70.63 (0.06)	70.67 (0.09)	71.27 (0.14)	72.73 (0.53)	73.66 (0.17)	74.17 (0.18)
Connect		11.83	12.4	13.66	15.63	19.53	27.66
(42,71.69%)	**DE$_{mirf}$**	71.57 (0.69)	71.78 (0.68)	72.53 (0.87)	73.47 (0.66)	73.98 (0.31)	73.95 (0.26)
		8.46	8.9	10.26	12.23	14.8	20.36
	Sig. Test	+	+	+	+	+	+
	DE$_{mi}$	86.00 (5.49)	85.11 (6.17)	86.88 (4.01)	86.44 (5.39)	86.11 (4.80)	86.55 (4.33)
Promoter		9.63	9.8	10.83	12.53	16.16	30.36
(57,90.00%)	**DE$_{mirf}$**	85.44 (6.52)	84.11 (5.51)	83.88 (5.87)	85.77 (4.62)	87.44 (4.34)	87.66 (3.05)
		9.06	9.8	10.53	11.23	14.6	28.23
	Sig. Test	=	=	−	=	=	=
	DE$_{mi}$	67.85 (3.27)	70.55 (4.17)	72.49 (3.71)	74.52 (2.03)	74.85 (1.85)	73.39 (1.40)
Splice		9.03	9.5	10.6	12	14.76	23.93
(60,66.77%)	**DE$_{mirf}$**	71.68 (4.17)	72.32 (3.95)	73.52 (3.62)	74.71 (2.67)	75.59 (2.21)	74.34 (1.34)
		9.56	9.46	11.13	11.9	14.06	20.2
	Sig. Test	+	+	=	=	+	+
	DE$_{mi}$	79.58 (5.48)	84.17 (3.12)	89.38 (2.34)	94.26 (1.02)	97.37 (0.40)	98.75 (0.12)
Optic		12.96	13.5	16.1	18.9	25.6	48.23
(64, 98.87%)	**DE$_{mirf}$**	89.12 (7.04)	91.34 (3.62)	90.51 (4.69)	94.17 (2.67)	97.63 (0.65)	98.57 (0.16)
		11.73	12.33	12.46	15.3	22.96	39.8
	Sig. Test	+	+	=	=	=	−
	DE$_{mi}$	64.25 (3.76)	63.84 (2.70)	64.61 (2.91)	64.56 (2.33)	64.20 (2.24)	63.53 (2.21)
Audiology		21.16	20.90	22.30	24.83	28.03	37.50
(68, 64.62%)	**DE$_{mirf}$**	72.00 (5.27)	67.38 (7.03)	70.10 (5.49)	68.30 (5.66)	65.17 (2.69)	64.82 (1.92)
		14.26	13.86	16.13	17.66	22.16	36.16
	Sig. Test	+	+	+	+	=	+
	DE$_{mi}$	93.47 (0.18)	93.58 (0.18)	93.58 (0.17)	93.59 (0.16)	93.68 (0.12)	93.74 (0.06)
Coil2000		30.33	30.33	32.93	36.1	43.13	58.33
(85,93.73%)	**DE$_{mirf}$**	93.69 (0.15)	93.71 (0.14)	93.63 (0.18)	93.71 (0.17)	93.65 (0.11)	93.80 (0.12)
		17.16	18.03	18.8	20.36	23.56	39.03
	Sig. Test	+	+	+	+	=	+
	DE$_{mi}$	81.13 (2.55)	81.42 (2.41)	82.82 (2.05)	83.38 (1.31)	82.90 (1.34)	82.31 (1.09)
DNA		57.63	57.50	60.26	65.53	73.06	95.83
(180,81.70%)	**DE$_{mirf}$**	81.57 (2.37)	81.07 (2.66)	82.47 (2.35)	83.69 (1.75)	83.26 (1.72)	83.14 (1.01)
		55.80	56.73	58.03	61.23	65.73	81.13
	Sig. Test	=	=	=	=	=	+
	DE$_{mi}$	70.40 (2.56)	70.85 (2.52)	71.27 (2.86)	72.82 (2.63)	72.77 (2.95)	75.08 (2.65)
PCMAC		1523.40	1523.76	1523.23	1524.53	1529.33	1552.26
(3289,70.10%)	**DE$_{mirf}$**	73.63 (2.60)	73.92 (2.32)	74.03 (2.33)	74.62 (2.38)	75.24 (1.98)	75.94 (1.62)
		1484.60	1487.53	1494.50	1499	1519.93	1619.60
	Sig. Test	+	+	+	+	+	=

the non-dominated solutions in the union set (referred as the best front) are also used for the comparison of the approaches.

The results of MODE$_{mirf}$, MODE$_{mi}$ and single objective approaches on the test sets are shown in Fig. 3, where each chart corresponds to the solutions of one dataset used in the experiments. In each chart, the horizontal axis represents the number of features, and the vertical axis represents the classification accuracy. On top of each chart, the numbers in the brackets correspond to the number of available features and the classification accuracy using all features. In charts, '-A' and '-B' represents the "average" and the "best" fronts, respectively. Single objective approaches may obtain the same feature subset size and same classification accurary in different runs in some datasets. Therefore, the plotted points on some charts for single objective approaches may be fewer than 30 distinct points.

According to Fig. 3, the average fronts of MODE$_{mi}$ (shown by MODE-MIFS-A) include a smaller number of features and achieve similar or higher classification performance than using all features

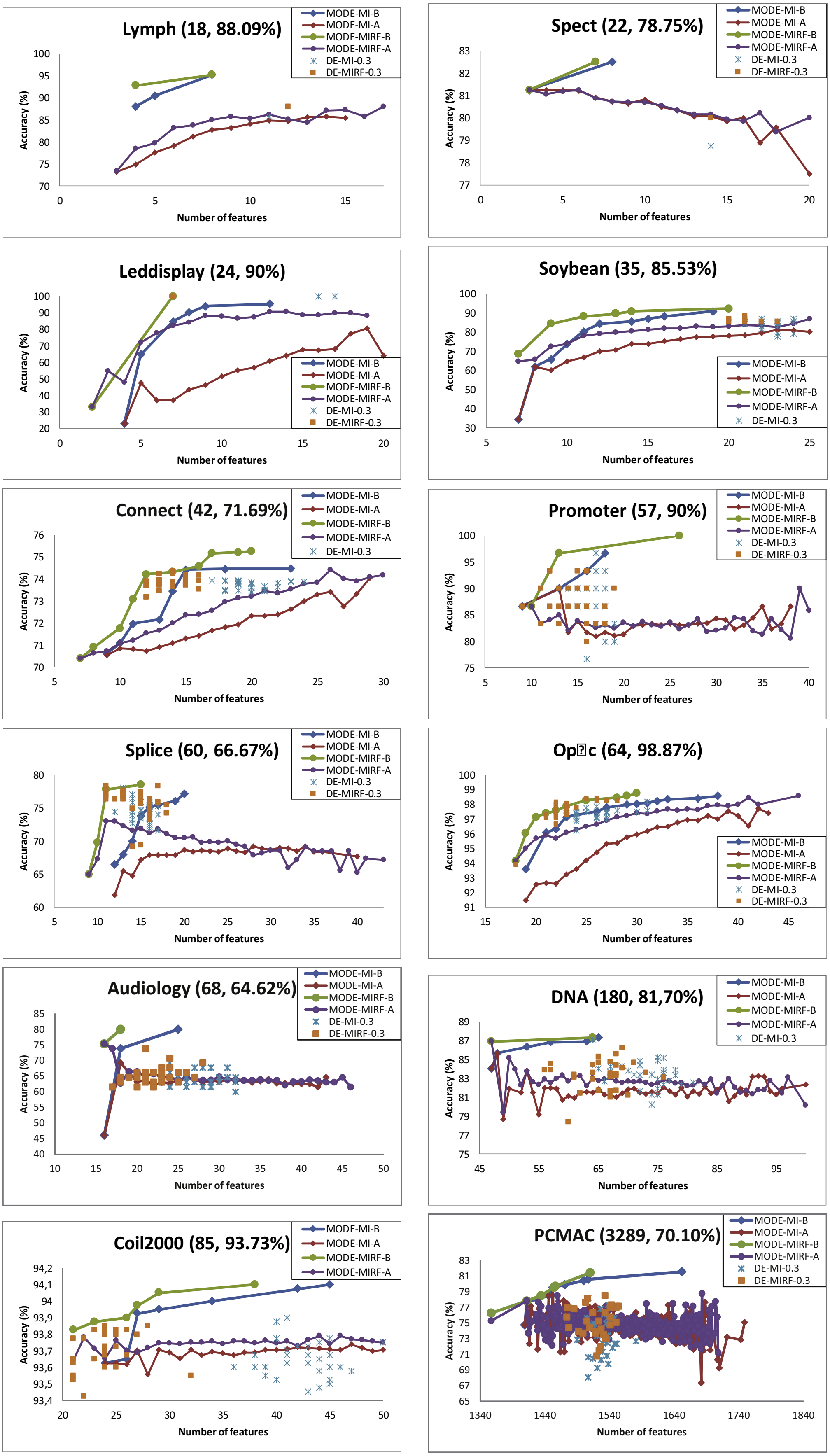

Fig. 3. Results of multi-objective approaches on test sets over KNN.

Table 3
Results of traditional approaches.

	Lymph			Spect			Leddisplay		
Method	CfsF	CfsB	GSBS	CfsF	CfsB	GSBS	CfsF	CfsB	GSBS
Accuracy	90.48	93.48	84.21	81.25	80	82.50	100	100	100
Size	9	9	24	12	10	18	7	7	5
	Soybean			Connect			Promoter		
Method	CfsF	CfsB	GSBS	CfsF	CfsB	GSBS	CfsF	CfsB	GSBS
Accuracy	85.53	85.53	84.21	70.73	70.73	71.68	90	90	90
Size	12	11	24	6	6	41	6	6	50
	Splice			Optic			Audiology		
Method	CfsF	CfsB	GSBS	CfsF	CfsB	GSBS	CfsF	CfsB	GSBS
Accuracy	72.83	72.83	68.65	98.69	98.69	98.75	64.62	64.62	64.62
Size	28	28	47	36	36	38	9	10	24
	Coil2000			DNA			PCMAC		
Method	CfsF	CfsB	GSBS	CfsF	CfsB	GSBS	CfsF	CfsB	GSBS
Accuracy	93.58	93.53	93.83	85.08	85.08	82.63	83.51	Null	Null
Size	10	20	31	34	34	173	47	Null	Null

except for the Lymph and Leddisplay datasets. Especially in the Leddisplay dataset, the average front performance of MODE$_{mi}$ is very low when compared to the classification performance obtained by using all features, but the classification performance in MODE-MIFS-B is high. Therefore, it can be inferred that for the solutions including the same number of features, there are a variety of combinations of feature subsets with different classification performance. It can be also inferred that in different runs, for the same feature subset size with the same fitness value (evaluated by Eq. (12)), MODE$_{mi}$ does not guarantee the same classification performance. In terms of the best fronts, MODE$_{mi}$ evolves the feature subsets that achieve higher classification performance than using all features in almost all cases. Especially in some cases which provides higher classification accuracy, MODE$_{mi}$ is able to eliminate at least 50% of available features. For instance, in the Soybean dataset, one non-dominated solution reduced the feature subset size from 35 to 16 and increased the classification accuracy from 85.53% to 88.15%. The results suggest that MODE$_{mi}$ can search the solution space and automatically evolve a set of feature subsets (solutions) to reduce the feature subset size and potentially increase the classification accuracy.

According to Fig. 3, the average fronts of MODE$_{mirf}$ (MODE-MIRFFS-A) are able to obtain the feature subsets providing similar or higher classification performance than using all features except for the Lymph dataset. It is also seen that the average fronts of MODE$_{mirf}$ get more feature subsets than MODE$_{mi}$ in terms of achieving better classification performance than using all features. As in MODE$_{mi}$, the non-dominated solutions of MODE$_{mirf}$ include a smaller number of features and achieve better classification performance than using all features in all datasets. In a significant number of MODE-MIRFFS-B solutions which provided better classification performance than using all features, the size of the feature subsets were reduced between 50% and 70% of available features. Therefore, MODE$_{mirf}$ can effectively explore the possible solution space to reduce the feature subset size and increase the classification accuracy.

Comparing MODE$_{mirf}$ with MODE$_{mi}$, it can be seen that MODE$_{mirf}$ outperforms MODE$_{mi}$ in terms of average fronts except for only some solutions of the Spect and Promoter datasets. The gap between MODE$_{mirf}$ and MODE$_{mi}$ can be easily observed, i.e., the lines representing the results of MODE$_{mi}$ mostly lay below the lines of MODE$_{mirf}$, indicating a lower classification than MODE$_{mirf}$. For instance, on the Leddisplay dataset, the feature subsets with 13 features get 90.71% average classification accuracy in MODE$_{mirf}$, but the average classification accuracy of the feature subsets with 13 features in MODE$_{mi}$ is only 60.80%. It is therefore not difficult to extract that the classification performance in MODE$_{mirf}$ does not vary widely for the solutions including the same number of features as in MODE$_{mi}$. The possible reason is that MODE$_{mirf}$ aimed to optimise three different criteria, which can capture different properties of the data to increase the classification performance consistently. Furthermore, MODE$_{mirf}$ is also superior to MODE$_{mi}$ in terms of the non-dominated solutions in almost all datasets. The comparisons show that both single objective and multi-objective DE approaches based on the proposed criterion can better explore the search space and achieve better solutions than the approaches based on the existing criterion.

5.3. Comparisons between multi-objective and single objective approaches

Comparing MODE$_{mi}$ with DE$_{mi}$ and DE$_{mirf}$, it is seen that in most cases, MODE$_{mi}$ (MODE-MIFS-B) eliminates irrelevant or redundant features more effectively and achieves better classification performance than DE$_{mi}$ and DE$_{mirf}$ with $\beta = 0.3$. When comparing MODE$_{mirf}$ with DE$_{mirf}$ and DE$_{mi}$, in almost all cases, MODE$_{mirf}$ (MODE-MIRRFS-B) also outperforms DE$_{mirf}$ and DE$_{mi}$ with $\beta = 0.3$ in terms of both the classification performance and the number of features. Therefore, considering both the MIFS and MIRFFS criteria in multi-objective design is more suitable and has more potential to explore the search space than single-objective design for feature selection problems. Furthermore, parameter β which keeps the balance between components does not need to be predefined in multi-objective design.

5.4. Comparisons with traditional approaches

Table 3 shows the results of the two traditional filter approaches (CfsF and CfsB) and one traditional wrapper approach (GSBS). The three traditional approaches produce a unique feature subset, so have a single accuracy for each test set. Note that it is not completely fair to compare filter approaches with wrapper approaches since wrappers use a classifier during the evaluation process.

Comparing single objective approaches (Table 2) with traditional filter approaches, it can be seen that DE$_{mi}$ achieves higher classification accuracy than traditional filter approaches in the Connect, Splice, Coil2000 and Audiology datasets. For the other datasets, traditional approaches outperform DE$_{mi}$. On the other hand, DE$_{mirf}$ performs similar or better classification accuracy than traditional approaches except for some cases. Comparing single objective approaches with the wrapper approach, GSBS, it is seen that single objective approaches outperform GSBS in all cases. Comparing multi-objective approaches (Fig. 3) with traditional filter approaches, it is seen that two multi-objective approaches select a smaller number of features and achieve higher classification performance than two traditional filter approaches except for the Promoter and Coil2000 datasets. Furthermore, multi-objective approaches outperform GSBS in all datasets in terms of the classification accuracy and the feature subset size.

5.5. Further comparisons

To further test the performance of the proposed algorithms, we compared the proposed both single objective method (DE$_{mirf}$) and multi-objective method (MODE$_{mirf}$) with six existing PSO based filter feature selection methods proposed in [64], including two single objective methods (*PSOMI* based on PSO and MIFS, and *PSOE* based on PSO and an entropy based information gain measure), and four multi-objective PSO methods (*NSfsMI* and *NSfsE* based on non-dominated sorting based multi-objective PSO [65] with MIFS and the entropy measures, respectively, and *CMDfsMI* and *CMDfsE* based on multi-objective PSO in [66] with MIFS and the entropy

Table 4
The CPU time results of single objective approaches.

Dataset	Method	β=1	β=0.9	β=0.7	β=0.5	β=0.3	β=0.1
Lymph	**DE$_{mi}$**	0.20 (0.02)	0.20 (0.02)	0.20 (0.02)	0.22 (0.03)	0.27 (0.03)	0.37 (0.03)
	DE$_{mirf}$	0.16 (0.03)	0.15 (0.02)	0.16 (0.02)	0.15 (0.02)	0.15 (0.01)	0.14 (0.02)
	Sig. Test	+	+	+	+	+	+
Spect	**DE$_{mi}$**	0.25 (0.03)	0.25 (0.02)	0.26 (0.02)	0.31 (0.02)	0.38 (0.03)	0.50 (0.04)
	DE$_{mirf}$	0.16 (0.02)	0.15 (0.02)	0.16 (0.01)	0.15 (0.02)	0.16 (0.03)	0.17 (0.02)
	Sig. Test	+	+	+	+	+	+
Leddisplay	**DE$_{mi}$**	0.28 (0.03)	0.28 (0.03)	0.31 (0.03)	0.33 (0.03)	0.41 (0.03)	0.51 (0.01)
	DE$_{mirf}$	0.16 (0.02)	0.16 (0.01)	0.15 (0.01)	0.17 (0.02)	0.16 (0.01)	0.16 (0.02)
	Sig. Test	+	+	+	+	+	+
Soybean	**DE$_{mi}$**	0.34 (0.03)	0.35 (0.02)	0.37 (0.03)	0.44 (0.03)	0.55 (0.03)	0.65 (0.03)
	DE$_{mirf}$	0.17 (0.02)	0.17 (0.01)	0.16 (0.02)	0.17 (0.02)	0.17 (0.02)	0.16 (0.02)
	Sig. Test	+	+	+	+	+	+
Connect	**DE$_{mi}$**	14.3 (2.29)	12.3 (3.01)	13.85 (0.64)	14.81 (1.04)	18.15 (1.11)	25.23 (1.87)
	DE$_{mirf}$	3.53 (0.77)	4.21 (2.93)	4.43 (3.01)	9.91 (4.38)	13.91 (0.92)	17.69 (1.73)
	Sig. Test	+	+	+	+	+	+
Promoter	**DE$_{mi}$**	0.48 (0.05)	0.48 (0.06)	0.49 (0.04)	0.50 (0.04)	0.58 (0.03)	0.79 (0.02)
	DE$_{mirf}$	0.17 (0.02)	0.17 (0.02)	0.16 (0.01)	0.18 (0.02)	0.16 (0.02)	0.17 (0.02)
	Sig. Test	+	+	+	+	+	+
Splice	**DE$_{mi}$**	0.52 (0.04)	0.52 (0.02)	0.52 (0.03)	0.56 (0.05)	0.67 (0.04)	0.84 (0.03)
	DE$_{mirf}$	0.21 (0.02)	0.22 (0.02)	0.23 (0.03)	0.21 (0.02)	0.22 (0.02)	0.22 (0.02)
	Sig. Test	+	+	+	+	+	+
Optic	**DE$_{mi}$**	0.65 (0.06)	0.64 (0.04)	0.67 (0.04)	0.73 (0.04)	0.86 (0.04)	1.34 (0.06)
	DE$_{mirf}$	0.26 (0.03)	0.26 (0.03)	0.27 (0.04)	0.27 (0.03)	0.30 (0.02)	0.38 (0.02)
	Sig. Test	+	+	+	+	+	+
Audiology	**DE$_{mi}$**	0.60 (0.07)	0.58 (0.07)	0.57 (0.06)	0.57 (0.04)	0.58 (0.07)	0.65 (0.03)
	DE$_{mirf}$	0.26 (0.03)	0.26 (0.02)	0.26 (0.02)	0.27 (0.05)	0.27 (0.05)	0.23 (0.02)
	Sig. Test	+	+	+	+	+	+
Coil2000	**DE$_{mi}$**	1.51 (0.08)	1.58 (0.09)	1.61 (0.09)	1.73 (0.10)	1.93 (0.07)	2.49 (0.09)
	DE$_{mirf}$	0.59 (0.06)	0.59 (0.03)	0.58 (0.03)	0.59 (0.04)	0.62 (0.04)	0.81 (0.05)
	Sig. Test	+	+	+	+	+	+
DNA	**DE$_{mi}$**	3.30 (0.15)	3.31 (0.06)	3.42 (0.08)	3.58 (0.07)	4.04 (0.11)	5.21 (0.25)
	DE$_{mirf}$	0.43 (0.03)	0.42 (0.02)	0.42 (0.04)	0.48 (0.05)	0.44 (0.03)	0.47 (0.03)
	Sig. Test	+	+	+	+	+	+
PCMAC	**DE$_{mi}$**	124.33 (7.68)	136.39 (3.17)	137.30 (5.09)	140.28 (4.56)	136.96 (16.42)	134.45 (7.57)
	DE$_{mirf}$	4.37 (0.29)	4.33 (0.18)	4.17 (0.15)	4.35 (0.22)	4.46 (0.10)	4.71 (0.18)
	Sig. Test	+	+	+	+	+	+

measures, respectively). The second multi-objective PSO framework [66] has shown to be better than the first one and other popular evolutionary multi-objective frameworks [67].

There are 5 datasets in common in this work and in [64], which are Lymph, Spect, Leddisplay, Soybean and Connect. When comparing the single objective methods, the proposed DE$_{mirf}$ achieves better performance than PSOMI and PSOE on four of the five datasets, with a slightly worse performance on Soybean than PSOMI and on Connect than PSOE but with a much smaller number of features. When comparing the multi-objective methods, the proposed MODE$_{mirf}$ achieves better performance than NSfsE and NSfsMI on four of the five datasets, and better than CMDfsMI and CMDfsE on three datasets, similar on one dataset, but worse on the other one dataset. Note that different data splitting may cause a slightly different accuracy on the dataset, but the superior performance of the new methods are significant, e.g nearly 10 percents accuracy increases. This is only a simple multi-objective DE framework, but the compared multi-objective PSO framework [66] is a sophisticated one. The above comparisons indicate that multi-objective DE with more advanced search mechanisms is very likely to have the potential of achieving even better performance, which confirms one of the motivations of this work.

5.6. Analysis of computational time

5.6.1. Comparisons of CPU time between DE$_{\mathrm{mirf}}$ and DE$_{\mathrm{mi}}$

The computational time results of single objective approaches are presented in terms of mean and standard deviation values over the 30 independent runs in Table 4. The standard deviation values are shown in brackets. The experiments are implemented in MATLAB2013a and are executed on a computer with an Intel Core i7-4700HQ 2.40 GHz CPU and 8 GB RAM. The results of Wilcoxon Rank Sum Test are shown via 'Sig. Test' as in Table 2, where '+' or '−' means that the computational time performance of DE$_{mirf}$ is shorter or longer than DE$_{mi}$ and '=' means that there is no significant change between DE$_{mirf}$ and DE$_{mi}$.

According to Table 4, the computational time of DE$_{mi}$ is increased inversely proportional to the β value, i.e., proportional to the feature subset size. The CPU time of DE$_{mi}$ for $\beta = 0.1$ is about two times as high as $\beta = 1$ in most cases. On the other hand, the computational time of DE$_{mirf}$ does not tend to increase inversely proportional to the β value, i.e., proportional to the feature subset size except for the Connect, Optic, Coil2000 and PCMAC datasets. The CPU time is increased in these datasets only between $\beta = 0.5$ and $\beta = 0.1$. Therefore, DE$_{mirf}$ can be treated as stable without no doubt in terms of the computational time.

Comparing DE$_{mirf}$ with DE$_{mi}$, it is seen that DE$_{mirf}$ can reduce the computational time at least a half or a quarter compared with DE$_{mi}$ in most cases. The computational time difference between DE$_{mirf}$ and DE$_{mi}$ is higher for the lower values of β. For instance, the gap between DE$_{mirf}$ and DE$_{mi}$ is increased from 0.31 to 0.62 s in the Promoter and Splice datasets, while the β value is decreased from 1 to 0.1. The results show that DE$_{mirf}$ achieves significantly better computational performance than DE$_{mi}$. That can be illustrated via 'Sig. Test' in Table 4. Therefore, DE$_{mirf}$ is superior to DE$_{mi}$ not only in terms of the classification performance and the number of features, but also in terms of the CPU computational time.

Table 5
The CPU time results of multi-objective approaches.

Dataset	MODE$_{mi}$	MODE$_{mirf}$	Sig. Test
Lymph	0.12 (0.01)	0.10 (0.01)	+
Spect	0.14 (0.01)	0.11 (0.01)	+
Leddisplay	0.39 (0.01)	0.27 (0.01)	+
Soybean	0.22 (0.01)	0.13 (0.01)	+
Connect	754.09 (37.57)	705.03 (25.05)	+
Promoter	0.31 (0.01)	0.10 (0.01)	+
Splice	2.42 (0.04)	2.29 (0.07)	+
Optic	8.43 (0.22)	7.56 (0.19)	+
Audiology	0.43 (0.05)	0.18 (0.03)	+
Coil2000	34.77 (0.81)	29.92 (0.66)	+
DNA	8.16 (0.37)	6.91 (0.35)	+
PCMAC	130.72 (27.46)	70.01 (9.95)	+

How can DE$_{mirf}$ complete the process in a shorter time in all cases and why cannot DE$_{mi}$ provide the stability in CPU computational time for different values of β? Given m selected features, as seen in Eq. (14), the time complexity of relevance and redundancy is $o(m)$ and $o(m^2)$, respectively; thus, the time complexity of DE$_{mi}$ is $o(m^2) + o(m) \approx o(m^2)$. On the other hand, the time complexity of relevance, ReliefF ranking and Fisher ranking is $o(m)$ as seen in Eq. (19); therefore, the time complexity of DE_{mirrfs} is about $o(m)$. Furthermore, DE$_{mirf}$ can remove/reduce irrelevant or redundant features more effectively than DE$_{mi}$, which also contributes to the improvement of the computational time.

5.6.2. Comparisons of CPU time between MODE$_{\text{mirf}}$ and MODE$_{\text{mi}}$

The computational time results of multi-objective approaches are presented in terms of mean and standard deviation values over the 30 independent runs in Table 5. The standard deviation values are shown in brackets. The experiments are implemented and executed on the same computer as in Section 5.2.1. The results of Wilcoxon Rank Sum Test are shown via 'Sig. Test', where '+' or '−' means the computational time performance of MODE$_{mirf}$ is shorter or longer than MODE$_{mi}$ and '=' means there is no significant difference between MODE$_{mirf}$ and MODE$_{mi}$.

According to Table 5, it is seen that MODE$_{mirf}$ can complete feature selection in a shorter time than MODE$_{mi}$ in all datasets, although the number of objectives in MODE$_{mirf}$ is higher than MODE$_{mi}$. The efficiency of MODE$_{mirf}$ is also supported by the Wilcoxon Rank Sum Test, which shows MODE$_{mirf}$ is significantly better than MODE$_{mi}$ in all datasets. How can MODE$_{mirf}$ be computationally more efficient? First, as mentioned in Section 5.2.1, the redundancy component of Eq. (14) increases the time complexity ($o(m^2)$) in MODE$_{mi}$. Furthermore, MODE [57] uses no complex and time consuming components to sort or renew individuals based on objective values like nondominated sorting genetic algorithm (NSGAII) [68] or multi objective particle swarm optimization (MOPSO) [69]. Instead of complex components such as non-dominated sorting and external archive, MODE uses multi-way greedy selection to renew or select individuals. Therefore, the computational time is not adversely affected by the number of objectives.

The comparisons confirm that both single objective and multi-objective DE approaches based on the proposed criterion can better explore the search space and achieve better solutions than the approaches based on the existing criterion. The comparisons also confirm to the fact that the proposed criterion (Eq. (19)) significantly improves the efficiency and effectiveness of both single objective and multi-objective DE algorithms in feature selection problems compared to the MIFS criterion (Eq. (14)).

5.6.3. Comparisons of CPU time with existing methods

When comparing with traditional methods, the forward selection method, i.e. CfsF, is much faster than the proposed methods, especially when the total number of features is small. CfsB following a backward selection method but with a filter measure is also faster than the proposed methods on small datasets, but slower than the proposed methods on large datasets, such as the PCMAC datasets, where both CfsB and GSBS cannot finish running within hours, but the proposed methods used minutes of time. The reason is that the backward selection method start with the full set of features, i.e. each evaluation involves a large dataset leading to a long computation time.

For making fair comparisons on CPU computational time, all approaches should be executed in computation environment, but in this work, we can indirectly compare the proposed multi-objective MODE$_{mirf}$ with the PSO based methods in [64]. The main reason is that when using EC methods for feature selection, the majority of the computational cost is used in the fitness evaluations. For (relatively) fair comparisons, different algorithms should use the same number of fitness evaluations. Since MODE$_{mirf}$ has shown to be faster than MODE$_{mi}$, and PSOMI, NSfsMI and CMDfsMI used the same fitness evaluation as MODE$_{mi}$, it is reasonable to say that MODE$_{mirf}$ is faster than PSOMI, NSfsMI and CMDfsMI. Furthermore, NSfsMI and CMDfsMI are much faster than PSOE, NSfsE and CMDfsE, which indicates that MODE$_{mirf}$ is faster than PSOE, NSfsE and CMDfsE. Of course, this is a general comparison on the computational cost, and the efficiency of all the algorithms can be improved in using a different programming language for implementation and a better computation environment.

6. Conclusions

The overall goal of this study was to develop new single objective and multi-objective DE based filter feature selection approaches to better searching for a set of feature subsets, which can eliminate irrelevant or redundant features and achieved better classification performance than using all features. This goal was successfully achieved by introducing a novel criterion inspired by feature ranking and mutual information, and adopting the most widely used criterion. Thus, two single objective (DE$_{mirf}$ and DE$_{mi}$) and two multi-objective (MODE$_{mirf}$ and MODE$_{mi}$) approaches were proposed for feature selection problems. The effectiveness of the approaches is demonstrated by comparing them to each other.

Experimental results show that in almost all cases, DE based on both the proposed and existing criteria can automatically evolve a small number of features and achieve better classification performance than using all features. Comparing the proposed and existing criteria, DE based on the proposed criterion outperformed the existing criterion in almost all cases in terms of both the number of features and the classification accuracy. Moreover, DE based on the proposed criterion searched the solution space much more efficiently than the existing criterion due to lower time complexity.

Experimental results also show that MODE based on both the proposed and existing criteria achieved similar or better classification performance than using all features and the single objective approaches in most datasets. Comparisons also indicate that MODE based on the proposed criterion outperformed the existing criterion in terms of both the best and the average fronts. Furthermore, the fluctuations on the classification performance among the solutions with the same number of features obtained by MODE based on the proposed criterion were lower than those produced by the existing criterion, which improved the performance of the average fronts. The computational time efficiency of the proposed criterion can be also illustrated in multi-objective approaches. Although the multi-objective design of the proposed criterion includes three objectives, it is also able to complete the feature selection process in a shorter time.

Instead of applying an existing criterion as an objective function which was mostly preferred in the literature, this paper proposes new DE-based approaches based on a novel criterion for filter based feature selection. The effectiveness and the efficiency of the approaches have been demonstrated in both single objective and multi-objective experimental studies. In future, we will further develop the multi-objective DE based filter approaches based on the proposed criterion to better explore the Pareto front of non-dominated solutions in feature selection and will try to redesign the proposed criterion for the continuous datasets.

Acknowledgments

This work was supported in part by the Marsden Fund of New Zealand Government under Contracts VUW1209, VUW1509 and VUW1615, Huawei Industry Fund E2880/3663, and the University Research Fund at Victoria University of Wellington 209862/3580, and 213150/3662.

Appendix. Further Comparisons Using Naive Bayes Classifier

To investigate whether the successful performance of the proposed single objective and multi-objective approaches can carry on other classification algorithms in addition to KNN. Naive Bayes

Table A.6
The results of single objective DE based on MIFS and MIRFFS over Naive Bayes.

Dataset	Method	β=1	β=0.9	β=0.7	β=0.5	β=0.3	β=0.1
Lymph **(18,90.47%)**	**DE**$_{mi}$	84.68 (1.20) 5.43	84.68 (1.21) 5.43	85.23 (0.97) 6.23	85.71 (4e-16) 8	85.71 (4e-16) 11	90.47 (0) 17
	DE$_{mirf}$	85.71 (4e−16) 5	85.71 (4e−16) 5	88.09 (3e−16) 8	88.09 (3e−16) 9	88.09 (3e−16) 12	90.47 (0) 17
	Sig. Test	+	+	+	+	+	=
Spect **(22,73.75%)**	**DE**$_{mi}$	81.25 (0) 8	81.25 (0) 8.1	81.25 (0) 10	81.25 (0) 12	81.25 (0) 14	78.75 (2e−16) 20
	DE$_{mirf}$	81.25 (0) 2	81.25 (0) 3	81.25 (0) 7	81.25 (0) 9	81.25 (0) 14	78.75 (2e−16) 22
	Sig. Test	=	=	=	=	=	=
Leddisplay **(24,99.33%)**	**DE**$_{mi}$	82.13 (0.27) 7.26	82 (2e−16) 9	93.33 (3e−16) 9.1	90 (4e−16) 12	99.70 (0.10) 16.1	99.33 (4e−16) 24
	DE$_{mirf}$	100 (0) 7	100 (0) 7	100 (0) 7	100 (0) 7	100 (0) 7	100 (0) 7
	Sig. Test	+	+	+	+	+	+
Soybean **(35,89.47%)**	**DE**$_{mi}$	63.99 (7.32) 9.56	64.95 (7.43) 10.13	71.97 (5.62) 12.7	81.62 (3.19) 16.6	90.87 (1.42) 22.53	89.12 (0.59) 33.88
	DE$_{mirf}$	70.96 (1.98) 10.46	71.53 (1.77) 10.8	72.32 (1.35) 13.3	81.53 (2.38) 15.96	86.35 (1.44) 21.33	88.85 (1.07) 27.93
	Sig. Test	+	+	=	=	−	=
Connect **(42,70.39%)**	**DE**$_{mi}$	70.39 (3e−16) 12.56	70.39 (3e−16) 12.06	70.39 (3e−16) 13.96	70.39 (3e−16) 15.46	70.39 (3e−16) 19.8	70.39 (3e−16) 27.86
	DE$_{mirf}$	70.39 (3e−16) 8.56	70.39 (3e−16) 8.8	70.39 (3e−16) 9.9	70.39 (3e−16) 11.9	70.39 (3e−16) 14.13	70.39 (3e−16) 20.06
	Sig. Test	=	=	=	=	=	=
Promoter **(57,80.00%)**	**DE**$_{mi}$	78.66 (7.03) 9.3	78.55 (4.84) 9.83	79.66 (5.41) 11	81.11 (6.27) 12.16	80 (4.28) 16.53	80 (3.60) 29.83
	DE$_{mirf}$	81.44 (4.68) 8.5	81.11 (6.27) 9.33	83.88 (5.47) 10.1	84 (5.70) 11.6	85.11 (4.35) 14.06	83.33 (4.46) 27.33
	Sig. Test	+	+	+	+	+	+
Splice **(60,75.97%)**	**DE**$_{mi}$	55.15 (5.92) 9.13	57.73 (5.79) 9.16	58.46 (6.14) 10.4	62.89 (2.78) 12.16	67.39 (2.04) 15.1	73.30 (1.02) 24.5
	DE$_{mirf}$	62.87 (4.75) 9.53	61.36 (5.81) 10.73	63.01 (5.82) 10.36	66.13 (5.06) 12.06	69.47 (2.69) 14.33	72.84 (1.16) 20.06
	Sig. Test	+	+	+	+	+	=
Optic **(64, 90.09%)**	**DE**$_{mi}$	72.31 (4.56) 13.43	76.16 (3.37) 14.1	79.58 (2.06) 15.66	84.96 (1.62) 18.3	88.17 (0.86) 25.63	90.01 (0.40) 49.83
	DE$_{mirf}$	77.53 (5.56) 11.93	76.88 (4.90) 11.36	79.72 (3.77) 12.43	83.57 (3.07) 15.53	87.86 (1.23) 22.13	89.88 (0.61) 40.1
	Sig. Test	+	+	=	=	=	=
Audiology **(68, 33.85%)**	**DE**$_{mi}$	72.76 (1.72) 20.96	73.33 (2.92) 21.5	72.56 (2.85) 21.8	73.23 (2.66) 25.3	71.43 (2.73) 28.66	64.71 (2.21) 38.1
	DE$_{mirf}$	69.48 (4.96) 13.66	69.79 (5.38) 13	70.25 (4.30) 15.33	69.33 (2.58) 17.73	70 (3.89) 20.96	60.05 (4.47) 35.4
	Sig. Test	−	−	=	−	=	−
Coil2000 **(85,74.48%)**	**DE**$_{mi}$	89.99 (1.49) 29.7	89.10 (2.42) 30.76	88.15 (1.85) 32.23	86.24 (2.05) 36.56	82.64 (2.23) 42.33	75.40 (0.51) 58.83
	DE$_{mirf}$	85.51 (4.75) 17.83	82.37 (4.26) 18.33	83.33 (4.8) 18.5	81.52 (4.57) 19.4	78.99 (2.84) 23.6	75.59 (0.73) 38.7
	Sig. Test	−	−	−	−	−	=
DNA **(180,91.23%)**	**DE**$_{mi}$	85.50 (1.93) 57.26	86.80 (2.01) 59.4	88.01 (1.64) 60.06	89.37 (1.41) 64.3	91.23 (1.05) 72.83	92.78 (0.51) 97.23
	DE$_{mirf}$	85.48 (2.45) 55.16	86.68 (2.16) 56.66	87.99 (1.86) 58.6	88.73 (2.01) 60.33	89.41 (1.80) 65.76	92.46 (0.74) 81.8
	Sig. Test	=	=	=	=	−	=
PCMAC **(3289,89.52%)**	**DE**$_{mi}$	83.93 (1.62) 1523.4	84.26 (1.97) 1523.76	84.61 (1.52) 1523.23	85.31 (1.49) 1524.53	85.24 (2.07) 1529.33	87.09 (1.32) 1552.26
	DE$_{mirf}$	85.26 (1.63) 1478	85.59 (1.44) 1480.86	85.97 (1.49) 1495.13	86.23 (1.63) 1495.33	86.81 (1.22) 1512.73	87.73 (0.97) 1621.83
	Sig. Test	+	+	+	+	+	+

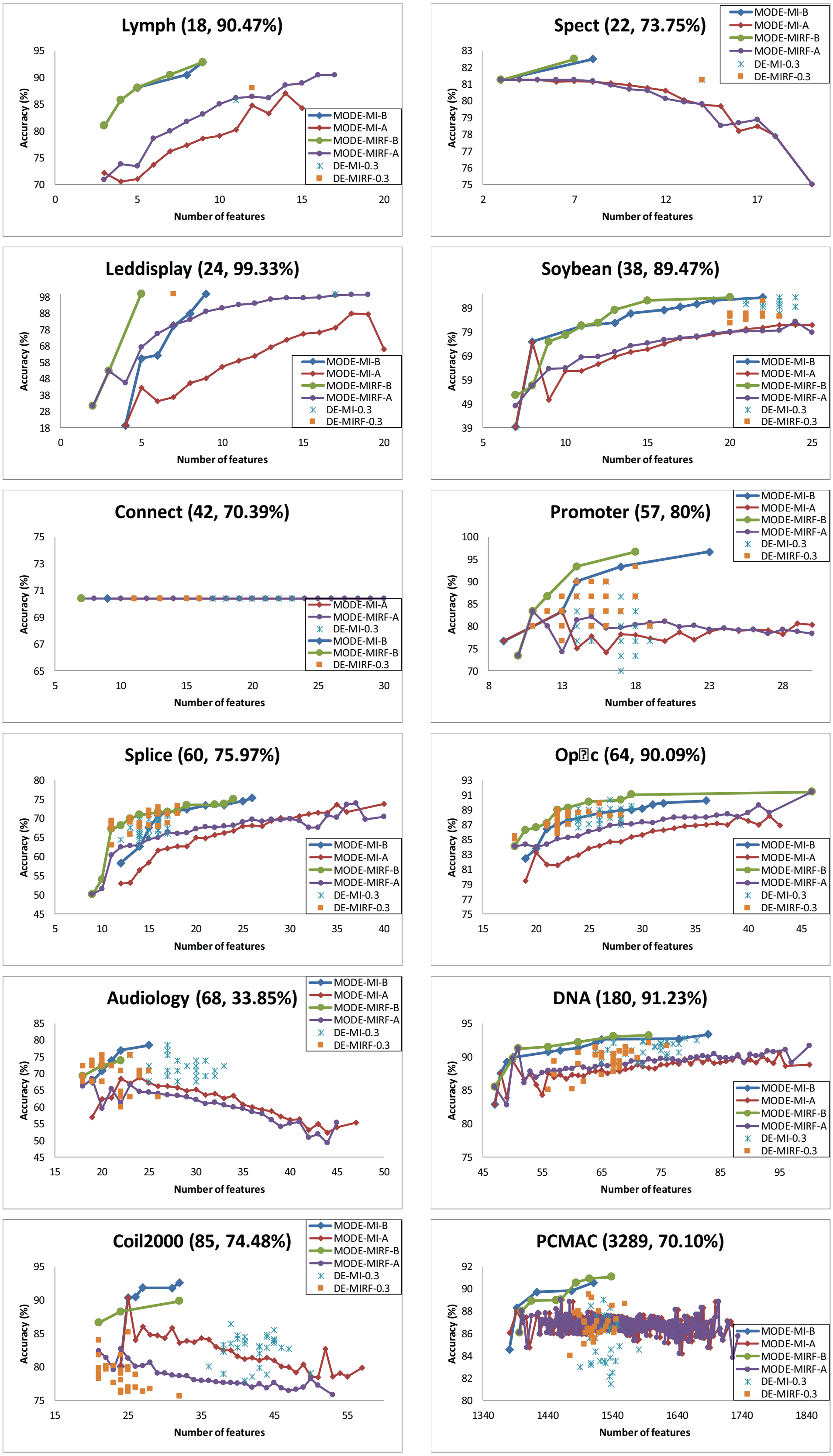

Fig. 4. Results of multi-objective approaches on test sets over Naive Bayes.

(NB) is used in the further experiments, since it is efficient, easy to implement and particularly useful for large datasets.

The results of single objective approaches are presented in Table A.6, which are obtained from 30 independent runs. The results include the classification performance, the feature subset size and the Wilcoxon Rank Sum Test. The numbers in the brackets underlying the caption of each dataset in Table A.6 represent the available number of features and the classification accuracy of NB using all features, respectively. According to Table A.6, both DE$_{mi}$ and DE$_{mirf}$ obtain generally similar or higher classification accuracy than using all features. Only in the Splice, Optic and PCMAC datasets, single objective approaches cannot achieve better classification accuracy than using all features, but the obtained classification accuracies in Optic are very close to the accuracy obtained by using all features. Comparing DE$_{mirf}$ with DE$_{mi}$, it can be observed that DE$_{mirf}$ outperforms DE$_{mi}$ in the Lymph, Leddisplay, Promoter, Splice and PCMAC datasets almost in all β values in terms of both the classification accuracy and the feature subset size. On the Spect, Soybean, Connect and Optic datasets, DE$_{mirf}$ mostly achieves similar classification performance using a smaller number of features than DE$_{mi}$. Overall, the results of the significance tests can show that the successful performance of DE$_{mirf}$ also carries on when using NB as a classifier.

The results of multi-objective approaches are presented with single objective approaches where $\beta = 0.3$ in Fig. 4. On top of each chart in Fig. 4, the numbers in brackets represent the feature subset size and the classification accuracy using NB with all features. The other concerning definitions and explanations related to charts can be found in Section 5.2. According to Fig. 4, both MODE$_{mi}$ and MODE$_{mirf}$ can automatically evolve a set of feature subsets yielding higher classification performance than using all features on all datasets. Especially in terms of the best fronts, high classification accuracies are achieved with less than 50% of the available features. For instance, on the Coil200 dataset, one best solution increased the classification accuracy from 74.48% to 92.52%, while the feature subset size was decreased from 85 to 32. Comparing multi-objective approaches with single objective approaches, it can be inferred from Fig. 4 that multi-objective approaches are more likely to find smaller feature subsets which achieves higher classification performance than DE$_{mi}$ on all the datasets and DE$_{mirf}$ except for the Audiology dataset. Accordingly, it is clear that both the MIFS and MIRFSS criteria in the multi-objective approach are able to search the possible solution space more effectively than single objective approaches in feature selection problems.

Comparing MODE$_{mirf}$ and MODE$_{mi}$, it can be observed from Fig. 4 that the best and average front lines of MODE$_{mi}$ are mostly lay below the lines of MODE$_{mirf}$ except for the Audiology and Coil2000 datasets. Furthermore, the gap between MODE$_{mirf}$ and MODE$_{mi}$ is extremely high, especially in terms of the average fronts. In other words, the classification performance of solutions with the same feature subset size obtained by MODE$_{mi}$ are more likely to vary, i.e., not stable and consistent compared to MODE$_{mirf}$. From the above comparisons, it can be concluded that considering the proposed criterion in both the single objective and multi-objective design can better search the possible solution space and obtain better solutions than the existing criterion in terms of the classification performance and the feature subset size over a different classification method.

References

[1] I. Guyon, S. Gunn, M. Nikravesh, L.A. Zadeh, Feature Extraction: Foundations and Applications (Studies in Fuzziness and Soft Computing), Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[2] A. Ekbal, S. Saha, Joint model for feature selection and parameter optimization coupled with classifier ensemble in chemical mention recognition, Knowl.-Based Syst. 85 (2015) 37–51.

[3] L. Jiang, H. Zhang, Z. Cai, A novel Bayes model: hidden Naive Bayes, IEEE Trans. Knowl. Data Eng. 21 (10) (2009) 1361–1371.

[4] C. Li, H. Li, One dependence value difference metric, Knowl.-Based Syst. 24 (5) (2011) 589–594.

[5] B. Xue, M. Zhang, W. Browne, X. Yao, A survey on evolutionary computation approaches to feature selection, IEEE Trans. Evolut. Comput. 20 (4) (2016) 606–626.

[6] P.E.M. Athanasios Tsanas Max A. Little, A simple filter benchmark for feature selection, J. Mach. Learn. Res. (2010) 1–24.

[7] G. Chandrashekar, F. Sahin, A survey on feature selection methods, Comput. Electr. Eng. 40 (1) (2014) 16–28.

[8] E. de la Hoz, E. de la Hoz, A. Ortiz, J. Ortega, A. Martínez-Álvarez, Feature selection by multi-objective optimisation: application to network anomaly detection by hierarchical self-organising maps, Knowl.-Based Syst. 71 (2014) 322–338.

[9] K.-J. Wang, K.-H. Chen, M.-A. Angelia, An improved artificial immune recognition system with the opposite sign test for feature selection, Knowl.-Based Syst. 71 (2014) 126–145.

[10] R. Li, J. Lu, Y. Zhang, T. Zhao, Dynamic adaboost learning with feature selection based on parallel genetic algorithm for image annotation, Knowl.-Based Syst. 23 (3) (2010) 195–201.

[11] S.S. Kannan, N. Ramaraj, A novel hybrid feature selection via symmetrical uncertainty ranking based local memetic search algorithm, Knowl.-Based Syst. 23 (6) (2010) 580–585.

[12] B. Xue, M. Zhang, W.N. Browne, A comprehensive comparison on evolutionary feature selection approaches to classification, Int. J. Comput. Intell. Appl. 14 (02) (2015) 1550008.

[13] R. Battiti, Using mutual information for selecting features in supervised neural net learning, IEEE Trans. Neural Netw. 5 (4) (1994) 537–550.

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

[15] N. Kwak, C.-H. Choi, Input feature selection for classification problems, IEEE Trans. Neural Netw. 13 (1) (2002) 143–159.

[16] D. Lin, X. Tang, Conditional infomax learning: An integrated framework for feature extraction and fusion, in: Computer Vision-ECCV 2006, in: Lecture Notes in Computer Science, 3951, Springer Berlin Heidelberg, 2006, pp. 68–82.

[17] L. Cervante, B. Xue, M. Zhang, L. Shang, Binary particle swarm optimisation for feature selection: a filter based approach, in: IEEE Congress on Evolutionary Computation (CEC'2012), 2012, pp. 881–888.

[18] H. Ge, T. Hu, Genetic algorithm for feature selection with mutual information, in: Proceedings of the Seventh International Symposium on Computational Intelligence and Design (ISCID '14), 2014, pp. 116–119.

[19] A. Al-ani, Ant colony optimization for feature subset selection, in: Proceedings of World Academy of Science, Engineering and Technology, 2005, pp. 35–38.

[20] M. Marinaki, Y. Marinakis, An island memetic differential evolution algorithm for the feature selection problem, in: Nature Inspired Cooperative Strategies for Optimization (NICSO'2013), in: Studies in Computational Intelligence, 512, Springer International Publishing, 2014, pp. 29–42.

[21] E. Hancer, B. Xue, M. Zhang, D. Karaboga, B. Akay, Pareto front feature selection based on artificial bee colony optimization, Inf. Sci. 422 (2018) 462–479.

[22] H. Wang, X. Jing, B. Niu, A discrete bacterial algorithm for feature selection in classification of microarray gene expression cancer data, Knowl.-Based Syst. 126 (2017) 8–19.

[23] B. Xue, L. Cervante, L. Shang, W.N. Browne, M. Zhang, Binary pso and rough set theory for feature selection: a multi-objective filter based approach, Int. J. Comput. Intell. Appl. 13 (02) (2014) 1450009.

[24] S. Das, P.N. Suganthan, Differential evolution: a survey of the state-of-the-art, IEEE Trans. Evolut. Comput. 15 (1) (2011) 4–31.

[25] B. Xue, M. Zhang, W.N. Browne, Multi-objective particle swarm optimisation (PSO) for feature selection, in: Proceeding of the 14th Annual Conference on Genetic and Evolutionary Computation Conference (GECCO), ACM, 2012, pp. 81–88.

[26] H. Li, Q. Zhang, Q. Chen, L. Zhang, Y.-C. Jiao, Multiobjective differential evolution algorithm based on decomposition for a type of multiobjective bilevel programming problems, Knowl.-Based Syst. 107 (2016) 271–288.

[27] M. Robnik-Sikonja, I. Kononenko, Theoretical and empirical analysis of relieff and rrelieff, Mach. Learn. 53 (1-2) (2003) 23–69.

[28] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, Inc., 1995.

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

[30] F. Rieke, D. Warland, R. de Ruyter van Steveninck, W. Bialek, Spikes: Exploring the Neural Code, MIT Press, Cambridge, MA, USA, 1999.

[31] F. Fleuret, Fast binary feature selection with conditional mutual information, J. Mach. Learn. Res. 5 (2004) 1531–1555.

[32] J. Benesty, J. Chen, Y. Huang, I. Cohen, Pearson correlation coefficient, in: Noise Reduction in Speech Processing, in: Springer Topics in Signal Processing, 2, Springer Berlin Heidelberg, 2009, pp. 1–4.

[33] X. He, D. Cai, P. Niyogi, Laplacian score for feature selection, In NIPS, MIT Press, 2005.

[34] R. Liu, N. Yang, X. Ding, L. Ma, An unsupervised feature selection algorithm: Laplacian score combined with distance-based entropy measure, in: Intelligent Information Technology Application, 2009. IITA 2009. Third International Symposium on, 3, 2009, pp. 65–68.

[35] L. Zhu, L. Miao, D. Zhang, Iterative Laplacian score for feature selection, in: C.-L. Liu, C. Zhang, L. Wang (Eds.), Pattern Recognition, Communications in Computer and Information Science, 321, Springer Berlin Heidelberg, 2012, pp. 80–87.
[36] M. Hall, Correlation-based feature selection for discrete and numeric class machine learning, 2000, pp. 359–366.
[37] P. Estevez, M. Tesmer, C. Perez, J. Zurada, Normalized mutual information feature selection, IEEE Trans. Neural Netw. 20 (2) (2009) 189–201.
[38] G. Brown, A new perspective for information theoretic feature selection, in: Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS'09), 5, Journal of Machine Learning Research - Proceedings Track, 2009, pp. 49–56.
[39] A. Al-Ani, M.D. (2002), A new technique for combining multiple classifiers using the dempster-shafer theory of evidence, J. Artif. Intell. Res. 17 (2002) 333–361.
[40] L. Zhang, L. Jiang, C. Li, A new feature selection approach to naive bayes text classifiers, Int. J. Pattern Recognit. Artif. Intell. 30 (02) (2016) 1650003.
[41] C. Freeman, D. Kuliýç, O. Basir, An evaluation of classifier-specific filter measure performance for feature selection, Pattern Recognit. 48 (5) (2015) 1812–1826.
[42] K. Yu, W. Ding, X. Wu, Lofs: A library of online streaming feature selection, Knowl.-Based Syst. 113 (2016) 1–3.
[43] Z. Chen, C. Wu, Y. Zhang, Z. Huang, B. Ran, M. Zhong, N. Lyu, Feature selection with redundancy-complementariness dispersion, Knowl.-Based Syst. 89 (2015) 203–217.
[44] F. Li, D. Miao, W. Pedrycz, Granular multi-label feature selection based on mutual information, Pattern Recognit. 67 (2017) 410–423.
[45] H. Bostani, M. Sheikhan, Hybrid of binary gravitational search algorithm and mutual information for feature selection in intrusion detection systems, Soft Comput. 21 (9) (2017) 2307–2324.
[46] R. Khushaba, S. Kodagoda, S. Lal, G. Dissanayake, Driver drowsiness classification using fuzzy wavelet-packet-based feature-extraction algorithm, IEEE Trans. Biomed. Eng. 58 (1) (2011) 121–131.
[47] E. Hancer, B. Xue, M. Zhang, D. Karaboga, B. Akay, A multi-objective artificial bee colony approach to feature selection using fuzzy mutual information, in: IEEE Congress on Evolutionary Computation (CEC), 2015, pp. 2420–2427.
[48] J. Huang, P. Rong, A hybrid genetic algorithm for feature selection based on mutual information, in: Information Theory and Statistical Learning, Springer US, 2009, pp. 125–152.
[49] H.B. Nguyen, B. Xue, I. Liu, M. Zhang, Filter based backward elimination in wrapper based pso for feature selection in classification, in: IEEE Congress on Evolutionary Computation (CEC'2014), 2014, pp. 3111–3118.
[50] H.B. Nguyen, B. Xue, P. Andreae, Mutual information for feature selection: estimation or counting? Evolut. Intell. 9 (3) (2016) 95–110.
[51] A. Al-ani, Ant colony optimization for feature subset selection, in: Proceedings of World Academy of Science, Engineering and Technology, 2005, pp. 35–38.
[52] R. Khushaba, A. Al-Ani, A. AlSukker, A. Al-Jumaily, A combined ant colony and differential evolution feature selection algorithm, in: Ant Colony Optimization and Swarm Intelligence, in: Lecture Notes in Computer Science, 5217, Springer Berlin Heidelberg, 2008, pp. 1–12.
[53] P. Moradi, M. Rostami, Integration of graph clustering with ant colony optimization for feature selection, Knowl.-Based Syst. 84 (2015) 144–161.
[54] B. Xue, L. Cervante, L. Shang, W.N. Browne, M. Zhang, Multi-objective evolutionary algorithms for filter based feature selection in classification, Int. J. Artif. Intell. Tools 22 (04) (2013) 1350024.
[55] A.K. Das, S. Das, A. Ghosh, Ensemble feature selection using bi-objective genetic algorithm, Knowl.-Based Syst. 123 (2017) 116–127.
[56] R. Kohavi, G.H. John, Wrappers for feature subset selection, Artif. Intell. 97 (1-22) (1997) 273–324.
[57] G. Reynoso-Meza, Multi-objective optimization differential evolution algorithm, 2012, http://cpoh.upv.es/en/research/software.html.
[58] K. Price, R.M. Storn, J.A. Lampinen, Differential evolution: a practical approach to global optimization, Springer Science & Business Media, 2006.
[59] J. Lampinen, Solving problems subject to multiple nonlinear constraints by differential evolution, in: 7th. Internation Conf. Soft Computing, 2001, pp. 50–57.
[60] K. Bache, M. Lichman, UCI machine learning repository, 2013, (????).
[61] M.A. Hall, Correlation-based Feature Subset Selection for Machine Learning, The University of Waikato, Hamilton, New Zealand, 1999 Ph.D. thesis.
[62] R. Caruana, D. Freitag, Greedy attribute selection, in: Proceedings of the Eleventh International Conference on Machine Learning, Morgan Kaufmann, 1994, pp. 28–36.
[63] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I.H. Witten, The weka data mining software: an update, SIGKDD Explor. 11 (2009) 931–934.
[64] B. Xue, L. Cervante, L. Shang, W.N. Browne, M. Zhang, A multi-objective particle swarm optimisation for filter based feature selection in classification problems, Connect. Sci. 24 (2-3) (2012) 91–116.
[65] X. Li, A non-dominated sorting particle swarm optimizer for multiobjective optimization, in: Proceedings of the 5th Annual Conference on Genetic and Evolutionary Computation (GECCO), Springer, 2003, pp. 37–48.
[66] M.R. Sierra, C.A.C. Coello, Improving pso-based multi-objective optimization using crowding, mutation and epsilon-dominance, in: International Conference on Evolutionary Multi-Criterion Optimization, Springer, 2005, pp. 505–519.
[67] B. Xue, M. Zhang, W.N. Browne, Particle swarm optimization for feature selection in classification: a multi-objective approach, IEEE Trans. Cybern. 43 (6) (2013) 1656–1671.
[68] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: Nsga-II, IEEE Trans. Evolut. Comput. 6 (2) (2002) 182–197.
[69] C. Coello, G. Pulido, M. Lechuga, Handling multiple objectives with particle swarm optimization, IEEE Trans. Evolut. Comput. 8 (3) (2004) 256–279.