International Journal on Artificial Intelligence Tools
Vol. 22, No. 4 (2013) 1350024 (31 pages)

DOI: 10.1142/S0218213013500243

World Scientific
www.worldscientific.com

# MULTI-OBJECTIVE EVOLUTIONARY ALGORITHMS FOR FILTER BASED FEATURE SELECTION IN CLASSIFICATION

BING XUE*,†, LIAM CERVANTE*, LIN SHANG†, WILL N. BROWNE*, MENGJIE ZHANG*

*School of Engineering and Computer Science, Victoria University of Wellington,
PO Box 600, Wellington 6140, New Zealand
†State Key Laboratory of Novel Software Technology, Nanjing University,
Nanjing 210046, China
{Bing.Xue, Liam.Cervante, Will.Browne, Mengjie.Zhang}@ecs.vuw.ac.nz,shanglin@nju.edu.cn

Received 25 July 2012
Accepted 29 May 2013
Published 19 August 2013

Feature selection is a multi-objective problem with the two main conflicting objectives of minimising the number of features and maximising the classification performance. However, most existing feature selection algorithms are single objective and do not appropriately reflect the actual need. There are a small number of multi-objective feature selection algorithms, which are wrapper based and accordingly are computationally expensive and less general than filter algorithms. Evolutionary computation techniques are particularly suitable for multi-objective optimisation because they use a population of candidate solutions and are able to find multiple non-dominated solutions in a single run. However, the two well-known evolutionary multi-objective algorithms, non-dominated sorting based multi-objective genetic algorithm II (NSGAII) and strength Pareto evolutionary algorithm 2 (SPEA2) have not been applied to filter based feature selection. In this work, based on NSGAII and SPEA2, we develop two multi-objective, filter based feature selection frameworks. Four multi-objective feature selection methods are then developed by applying mutual information and entropy as two different filter evaluation criteria in each of the two proposed frameworks. The proposed multi-objective algorithms are examined and compared with a single objective method and three traditional methods (two filters and one wrapper) on eight benchmark datasets. A decision tree is employed to test the classification performance. Experimental results show that the proposed multi-objective algorithms can automatically evolve a set of non-dominated solutions that include a smaller number of features and achieve better classification performance than using all features. NSGAII and SPEA2 outperform the single objective algorithm, the two traditional filter algorithms and even the traditional wrapper algorithm in terms of both the number of features and the classification performance in most cases. NSGAII achieves similar performance to SPEA2 for the datasets that consist of a small number of features and slightly better results when the number of features is large. This work represents the first study on NSGAII and SPEA2 for filter feature selection in classification problems with both providing field leading classification performance.

*Keywords*: Feature selection; evolutionary algorithms; multi-objective optimisation; filter approaches; genetic algorithms.

## 1. Introduction

Classification is one of the major tasks in machine learning and data mining, which involves the prediction of the class label for each instance according to the information described by its features. However, classification problems usually include a large number of features. Irrelevant and redundant features may reduce the classification performance due to the unnecessarily large search space. Feature selection aims to select a subset of relevant features to achieve similar or even better classification performance.[1–3] By selecting only the relevant features for classification, feature selection can reduce the running time, simplify the learned classifier, and/or increase the classification performance.[1,3]

Feature selection is a difficult problem because of two main reasons. The first reason is that there can be complex interaction between features. An individually relevant (irrelevant) feature may become redundant (relevant) when working together with other features. An optimal feature subset should be a group of complementary features that span over the diverse properties of the classes to properly discriminate them. The second reason is that the search space is large that is $2^n$ for $n$ features. So in most situations, it is impractical to conduct an exhaustive search for feature selection.[4,3] Therefore, feature selection algorithms need two key factors: an evaluation criterion, which determines the goodness of the selected feature subset, and a search technique, which searches the space of solutions to find the optimal feature subset.

Based on the evaluation criterion, existing feature selection approaches can be broadly classified into two categories: wrapper approaches and filter approaches. Wrapper approaches include a learning/classification algorithm as part of the evaluation function to determine the goodness of the selected feature subsets. Wrappers can often achieve better results than filter approaches, but the main drawbacks are their high computational cost and loss of generality.[4] Filter approaches use statistical characteristics of the data for evaluation and the feature selection search process is independent of a learning/classification algorithm. Compared with wrappers, filter approaches are argued to be computationally less expensive and more general.[1]

A variety of search techniques have been applied to feature selection such as greedy search.[5,6] However, most of the existing feature selection methods still suffer from different problems, such as stagnation in local optima and high computational cost.[3,7] In order to better address feature selection problems, an efficient global search technique is needed. Evolutionary computation algorithms, such as genetic algorithms (GAs),[8] genetic programming (GP)[9] and particle swarm optimisation (PSO),[3] are well-known for their global search ability and they have been applied to feature selection problems.

Most of the existing feature selection algorithms are wrapper approaches. Wrappers are less general and computational more expensive than filter approaches. Meanwhile, feature selection problems have the two main objectives of minimising both the classification error rate and the number of features. These two objectives

are usually conflicting to each other. Therefore, the optimal solution needs to be chosen in the presence of a trade-off between the two objectives. However, most of the existing algorithms are single objective methods. Evolutionary algorithms seem particularly suitable to solve multi-objective problems, because they simultaneously deal with a population of candidate solutions, which allows them to find multiple non-dominated solutions in a single run. Evolutionary multi-objective algorithms, such as non-dominated sorting based multi-objective genetic algorithm II (NSGAII)[10] and strength Pareto evolutionary algorithm 2 (SPEA2),[11] have been widely used in many areas.[12] However, the use of NSGAII and SPEA2 in *filter* based feature selection has not been investigated to date. Although mutual information and entropy as effective information measures have already been investigated by many researchers, they have never been used with NSGAII or SPEA2 for *multi-objective filter* feature selection. The work represents the first study on NSGAII and SPEA2 for filter feature selection in classification problems.

### 1.1. *Goals*

The overall goal of this paper is to develop a multi-objective, filter based feature selection approach to classification based on information theory and evolutionary multi-objective techniques to search for a set of non-dominated solutions (feature subsets), which contain a small number of features and achieve similar or even better classification performance than using all features. To achieve this goal, we will develop two information measurements (mutual information and entropy) and two multi-objective feature selection frameworks based on NSGAII and SPEA2. Thus four multi-objective feature selection algorithms will be proposed by applying the two information measurements to the two frameworks. These proposed feature selection algorithms will be examined and compared with three traditional feature selection methods and a single objective GA on eight benchmark problems of varying difficulty. Specifically, we will investigate

- whether the single objective GA approach with the two information measurements can select a small number of features and improve the classification performance over using all features;
- whether NSGAII based multi-objective feature selection algorithms can evolve a smaller number of features and achieve better classification performance than the single objective approach;
- whether SPEA2 based multi-objective feature selection algorithms can evolve a set of good feature subsets and outperform the single objective algorithm; and
- whether the proposed multi-objective algorithms can outperform the three traditional feature selection methods.

### 1.2. *Organisation*

The remainder of the paper is organised as follows. Section 2 provides background information of multi-objective optimisation, evolutionary computation techniques,

and related work on feature selection. Section 3 describes the proposed multi-objective feature selection algorithms, which are based on NSGAII and SPEA2 with mutual information and entropy. Section 4 presents the experimental design. The experimental results and discussions are provided in Section 5. Section 6 describes conclusions and future work.

## 2. Background

This section provides background about multi-objective optimisation, evolutionary techniques and also reviews typical related work on feature selection.

### 2.1. *Multi-objective optimisation*

Most optimisation problems naturally have multiple objectives and these objectives are normally conflicting with each other. Multi-objective optimisation seeks to simultaneously optimise (minimise or maximise) multiple conflicting objective functions. In general, a $k$-objective minimisation problem can be formulated as follows:

$$minimise \;\; F(x) = [f_1(x), f_2(x), \ldots, f_k(x)] \tag{1}$$

subject to:

$$g_i(x) \leq 0\,, \qquad i = 1, 2, \ldots, m \tag{2}$$

$$h_i(x) = 0\,, \qquad i = 1, 2, \ldots, l \tag{3}$$

where $f_i(x)$ is an objective function, which depends upon a vector of decision variables shown by $x$, $g_i(x)$ and $h_i(x)$ are the constraint functions.

When $k = 1$, the above model describes a single objective problem and the optimal solution is the one minimising the objective. However, when $k > 1$ (multi-objective problem), the quality of a solution is explained in terms of trade-offs between the $k$ conflicting objectives.

Let $y$ and $z$ be two solutions of the above $k$-objective minimisation problem. If the following conditions are met, one can say that $y$ dominates $z$ (or $z$ is dominated by $y$):

$$\forall\, i : f_i(y) \leq f_i(z) \quad \text{and} \quad \exists\, j : f_j(y) < f_j(z) \tag{4}$$

where $i, j \in \{1, 2, 3, \ldots\, k\}$. When a solution is not dominated by any other solutions, it is referred as a Pareto optimal solution or a non-dominated solution. The set of all Pareto optimal solutions forms the trade-off surface in the search space, the *Pareto front*. A multi-objective algorithm is designed to search for a set of non-dominated solutions.

Feature selection has the two main conflicting objectives, which are minimising both the number of features and the classification error rate. Therefore, feature selection can be expressed as a two-objective minimisation problem.

### 2.2. *Evolutionary algorithms*

Evolutionary computation is an area of artificial intelligence that covers the majority of the techniques inspired by principles of biological evolution.[13] Evolutionary techniques have been successfully applied to solve a variety of real-world problems.[13]

Genetic algorithms (GAs) are a typical evolutionary technique.[14] In a GA, each candidate solution is encoded as an individual, or a chromosome in the population. The evolutionary process of a GA usually starts from a population of randomly generated individuals. Based on the Darwinian principle of "survival of the fittest", the GA evolves toward the optimal solution in a series of generations. In each generation, the fitness of each individual is evaluated. According to their fitness, multiple individuals are selected from the current population and modified by performing genetic operators, such as crossover and mutation, to form a new population, which is used in the next generation. Generally, the GA terminates when either a maximum number of generations has been performed, or a satisfactory fitness level has been reached.

Evolutionary algorithms seem particularly suitable to solve multi-objective optimisation problems, because they simultaneously deal with a set of candidate solutions (the so-called population). This allows the algorithms to find multiple possible members of the Pareto optimal set in a single run, instead of having to perform different runs as in the case of the traditional mathematical programming techniques. In recent years, many evolutionary multi-objective algorithms have been developed. Two well-known algorithms are NSGAII and SPEA2, which have been successfully applied to a variety of areas.[10,11]

### 2.3. *Entropy and mutual information*

In information theory, entropy and mutual information can measure the information of random variables.[15] For example, let $X$ be a random variable with discrete values, its uncertainty can be measured by entropy $H(X)$ defined as

$$H(X) = -\sum_{x\in\mathcal{X}} p(x)\log_2 p(x) \tag{5}$$

where $p(x)$ is the probability density function of $X$.

For two random variables $X$ and $Y$ with their probability density function $p(x, y)$, the joint entropy $H(X, Y)$ is defined as

$$H(X,Y) = -\sum_{x\in\mathcal{X},y\in\mathcal{Y}} p(x,y)\log_2 p(x,y)\,. \tag{6}$$

When a variable is known and others are unknown, the remaining uncertainty is measured by the conditional entropy. Given $Y$, the conditional entropy $H(X|Y)$ of $X$ with respect to $Y$ is

$$H(X|Y) = -\sum_{x\in\mathcal{X},y\in\mathcal{Y}} p(x,y)\log_2 p(x|y) \tag{7}$$

where $p(x|y)$ is the posterior probabilities of $X$ given $Y$. $H(X|Y) = 0$ means that $X$ completely depends on $Y$ and no more other information is required to describe $X$ when $Y$ is known. $H(X|Y) = H(X)$ denotes that knowing $Y$ will do nothing to observe $X$.

Mutual information defines the information shared between two random variables. Given variable $X$, mutual information $I(X;Y)$ is how much information one can gain about variable $Y$.

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ &= H(Y) - H(Y|X) \\ &= -\sum_{x\in\mathcal{X}, y\in\mathcal{Y}} p(x,y)\log_2 \frac{p(x,y)}{p(x)p(y)}\,. \end{aligned} \tag{8}$$

According to Eq. (8), the mutual information $I(X;Y)$ will be large if two variables $X$ and $Y$ are closely related. $I(X;Y) = 0$ if $X$ and $Y$ are totally unrelated.

Let $c$ be a single discrete variable and $Z$ be a set of discrete variables. The information gain of $c$ given by $Z$ can be calculated as follows[16]:

$$\begin{aligned} IG(c|Z) &= H(c) - H(c|Z) \\ &= H(c) - (H(c\cup Z) - H(Z)) \\ &= H(c) + H(Z) - H(c\cup Z) \end{aligned} \tag{9}$$

where $H(Z)$ is the joint entropy of all the features in $Z$. If $Z = A, B, C$, then

$$H(A,B,C) = -\sum_{a\in A}\sum_{b\in B}\sum_{c\in C} p(abc)\log_2 p(abc)\,.$$

### 2.4. *Related work on feature selection*

A number of feature selection algorithms have been proposed in recent years.[1] Typical feature selection algorithms are reviewed in this section.

#### 2.4.1. *Traditional feature selection approaches*

The Relief algorithm[17] is a classical filter feature selection algorithm. Relief assigns a weight to each feature to denote the relevance of the feature to the class labels. However, Relief does not deal with redundant features, because it attempts to find all relevant features regardless of the redundancy between them. As decision trees (DT) use only relevant features that are required to completely classify the training set and remove all other features, Cardie[18] proposed a filter feature selection algorithm that used a DT to select a subset of features for a k-nearest neighbourhood algorithm (KNN). The FOCUS algorithm,[19] a filter algorithm, exhaustively examines all possible feature subsets, then selects the smallest feature subset. However, the FOCUS algorithm is computationally inefficient because of the exhaustive search.

Two commonly used wrapper feature selection methods are sequential forward selection (SFS)[5] and sequential backward selection (SBS).[6] SFS (SBS) starts with no features (all features), then candidate features are sequentially added to (removed from) the initial feature subset until the further addition (removal) does not increase the classification performance. The limitation of these two methods are that once a feature is selected (eliminated) it cannot be eliminated (selected) later, which is so-called nesting effect.[20] This limitation can be overcome by combining both SFS and SBS into one algorithm. Therefore, the "plus-$l$-take away-$r$" method was proposed by Stearns.[21] "plus-$l$-take away-$r$" performs $l$ times forward selection followed by $r$ times backward elimination. The challenge is to determine the optimal values of ($l$, $r$). To address this challenge, two floating feature selection algorithms were proposed by Pudil *et al.*,[22] namely sequential forward floating selection (SFFS) and sequential backward floating selection (SBFS). SFFS and SBFS were developed to automatically determine the values for ($l$, $r$). These two floating methods are regarded to be at least as good as the best sequential method, but they also suffer from the problem of stagnation in local optima.[20]

#### 2.4.2. *Evolutionary computation techniques for feature selection*

Recently, evolutionary techniques have gained more attention for solving feature selection problems. These include GAs, GP, PSO and ant colony optimisation (ACO).

Based on GAs, Huang and Wang[23] proposed a feature selection algorithm, which was used to simultaneously search for the best feature subset and optimise the kernel parameters in a support vector machine (SVM). Experimental results show that the proposed GA based algorithm outperformed a traditional parameters searching method, the Grid algorithm, in terms of both the number of features and the classification performance. Hamdani *et al.*[24] developed a multi-objective, *wrapper* feature selection algorithm using NSGAII, where the two objectives were the minimisation of both the number of features and the classification error rate. However, the performance of this algorithm was not compared with any other feature selection algorithm. Later, Soto *et al.*[25] also developed a *wrapper* based multi-objective feature selection algorithm, where NSGAII and SPEA2 were used as the search technique and four different learning algorithms were used in the experiments to evaluate the classification performance of the selected features. Guillén *et al.*[26] used NSGAII and local search to develop a memetic algorithm based multi-objective method for *wrapper* based multi-objective feature selection and simultaneously evolving Radial Basis Function Neural Networks (RBFNNs). In 2010, Huang *et al.*[27] developed a *wrapper* based multi-objective feature selection algorithm for customer churn prediction in telecommunications by using a modified NSGAII. In this approach, the true positive rate, true negative rate and the overall classification rate are used as the three objectives in NSGAII. Different from the above multi-objective algorithms, the number of features are not one of the objectives in NSGAII. This algorithm was examined on one churn pre diction dataset in telecommunications and achieved good classification performance with a small number of features. However, all these

multi-objective algorithms are *wrapper* based approaches, and there is no much work conducted on using NSGAII for *multi-objective filter* based feature selection. In this paper, we aim to develop a *filter* based *multi-objective* feature selection approach.

Memetic algorithms usually combine GAs and local search. Zhu *et al.*[28] proposed a hybrid wrapper and filter feature selection algorithm (WFFSA) based on a memetic algorithm. In WFFSA, a GA adds or deletes a feature based on the ranked individual features. Experiments show that WFFSA outperformed GAs and other methods. However, the performance of WFFSA may be limited when dealing with problems with high feature interaction, because features are ranked individually without considering the interaction between them.

Based on GP, Kourosh and Zhang[9] proposed a GP relevance measure (GPRM) to evaluate and rank subsets of features, and GPRM is also efficient in terms of feature selection. Muni *et al.*[29] developed a multi-tree GP algorithm for feature selection (GPmtfs) to simultaneously select a feature subset and design a classifier using the selected features. For a $c$-class problem, each classifier in GPmtfs has $c$ trees. Comparisons suggest GPmtfs achieved better results than SFS, SBS and other methods. However, the number of features selected increases when there are (synthetically added) noisy features.

Kourosh and Zhang[30] proposed a GP based filter approach to feature selection in binary classification problems. Unlike most filter methods that usually could only measure the relevance of a single feature to the class labels, the proposed algorithm can discover the hidden relationships between subsets of features and the target classes. Experiments show that the proposed algorithm improved the classification performance of classifiers while decreased their complexity. However, the proposed method might not be quite appropriate for the problems where the best feature subset is expected to have a very large number of features.

PSO has been applied to feature selection problems. Wang *et al.*[31] proposed a filter feature selection algorithm based on an improved binary PSO and rough sets theory.[32] The goodness of a particle is assigned as the relevance degree between class labels and selected features, which is measured by rough sets. This work also shows that the computation of the rough sets consumes most of the running time, which is a drawback of using rough sets in feature selection problems. Based on PSO, Esseghir *et al.*[33] proposed a filter-wrapper feature selection method, which aims to integrate the strengths of both filters and wrappers. The proposed filter-wrapper scheme encodes the position of each particle with a score, which reflects feature-class dependency levels evaluated by a predefined filter criterion. The fitness of a particle is the classification accuracy achieved by the selected features. Experimental results show that the proposed method achieved slightly better performance than a PSO based filter algorithm. As the proposed approach uses the wrapper scheme, it would be necessary to compare the work directly with a wrapper approach in order to judge its efficacy worth.

Unler and Murat[3] proposed a wrapper feature selection algorithm with an adaptive selection strategy, where a feature is chosen not only according to the likelihood

calculated by PSO, but also to its contribution to the features already selected. Experiments suggest that the proposed method outperformed the tabu search and scatter search algorithms. Lin *et al.*[34] proposed a wrapper feature selection algorithm to optimise the kernel parameters in SVM and search for the optimal feature subset simultaneously. Experimental results show that the proposed algorithm achieved slightly better performance than the GA-based algorithm developed by Huang and Wang.[23] Liu *et al.*[7] introduced a multi-swarm PSO (MSPSO) algorithm to search for the optimal feature subset and optimise the parameters of SVM simultaneously. Experiments show that the proposed feature selection method could achieve higher classification accuracy than grid search, standard PSO and GA. However, the proposed algorithm is computationally more expensive than the other three methods because of the large population size and complicated communication rules between different subswarms.

ACO as an evolutionary algorithm has also been applied to feature selection problems. Ming[35] proposed a feature selection method based on ACO and rough sets. The proposed algorithm starts with the features included in the core of the rough sets. Forward selection was adopted to search for the best feature subset. Experimental results showed that the proposed algorithm achieved better classification performance with fewer features than a C4.5 based feature selection algorithm. However, experiments did not compare the proposed method with other evolutionary based feature selection algorithms. Sivagaminathan *et al.*[36] applied ACO to a wrapper feature selection algorithm, where an artificial neural network (ANN) was used to evaluate the classification performance. Experimental results show that the proposed algorithm selected a small number of features and achieved better classification performance than using all features in most cases. Gao *et al.*[37] proposed an ACO based wrapper feature selection algorithm to network intrusion detection. However, only one problem was tested in the experiment, which does not demonstrate the robustness, scalability, or general applicability of the proposed technique.

In summary, different techniques have been applied to feature selection. Many studies have shown that evolutionary algorithms are efficient techniques for feature selection problems. However, most of the existing feature selection algorithms are wrapper approaches, which are computationally more expensive and less general than filter approaches. A relatively small number of filter feature selection approaches have been proposed in which rough sets and fuzzy sets theories are mainly used to evaluate the fitness of the selected features. However, Wang *et al.*[31] has already shown the drawback of high computational cost of using rough sets. Moreover, there are rare studies on multi-objective evolutionary technique for filter feature selection. Therefore, the investigation of an evolutionary multi-objective algorithm for filter based feature selection is still an open issue.

## 3. Proposed Multi-Objective Feature Selection Approaches

In this section, two filter criteria based on mutual information and entropy[16] are firstly described in this section. Two single objective benchmark feature selection

algorithms are developed based on each of the two filter criteria and a single objective GA. Then we propose two new multi-objective feature selection frameworks that form the new algorithms to treat feature selection as a multi-objective problem with the goal of minimising the number of features and maximising the relevance between the selected features and the class labels.

### 3.1. *Single objective algorithms based on GAs, mutual information and entropy*

Two single objective feature selection algorithms are firstly developed as baselines to test the performance of multi-objective algorithms, which will be proposed in this paper.

#### 3.1.1. *GAs and mutual information: GAMI*

Mutual information in information theory shows the relevance between two random variables. In classification problems, categorical features and the class labels can be treated as discrete variables. Therefore, mutual information can be used in feature selection. The relevance of a feature subset to the class labels can be evaluated by summing up the relevance of all individual features in the subset. However, this sum will be maximised when all the features are included. In order to reduce the number of features selected, the redundancy of the feature subset needs to be minimised, which can be shown by the mutual information between features in the subset. Based on mutual information, we proposed a filter fitness function for feature selection in an attempt to maximise the relevance between features and class labels and minimise the redundancy among features, which is shown in Eq. (10).[16] In this work, by using Eq. (10) as the fitness function and a GA as the search technique, we propose a filter feature selection algorithm (GAMI). This measure (Eq. (10)) was originally applied to a PSO algorithm and GAMI is its first application in a GA.

$$F_1 = Rel_1 - Red_1 \tag{10}$$

where

$$Rel_1 = \sum_{x \in X} I(x; c)\,, \qquad \text{and} \qquad Red_1 = \sum_{x_i, x_j \in X} I(x_i, x_j)$$

where $X$ stands for the selected feature subset and $x$ is a single feature in $X$. $c$ is the class labels. $I(x; c)$ and $I(x_i, x_j)$ can be calculated according to Eq. (8). $Rel_1$ determines the relevance of the selected feature subset and $Red_1$ shows the redundancy contained in the selected feature subset. $F_1$ aims to maximise the relevance $Rel_1$ and simultaneously minimise the redundancy $Red_1$ in the selected feature subset.

In GAMI, each individual (chromosome) in the population represents a subset of features. For a $n$-dimensional feature search space, each individual is encoded by a $n$-bit binary string. The bit with value "1" indicates the feature is selected in the subset, and "0" otherwise.

3.1.2. *GAs and entropy*: *GAE*

Mutual information can find the two-way relevance and redundancy between features, which are caused by feature interaction. However, it could not handle multi-way complex feature interaction, which is one of the challenges in feature selection. Entropy in information theory can measure the relevance between a group of features based on which, we proposed another evaluation criterion to discover multi-way relevance and redundancy among features and the fitness function can be seen in Eq. (11).[16] In this work, by using Eq. (11) as the fitness function and a GA as the search technique, we propose a filter feature selection algorithm (GAE).

$$F_2 = Rel_2 - Red_2 \tag{11}$$

where

$$Rel_2 = IG(c|X) \quad \text{and} \quad Red_2 = \frac{1}{|S|}\sum_{x\in X} IG(x|\{X/x\})$$

where $X$, $x$ and $c$ have the same meanings as in Eq. (10). $IG(c|X)$ and $IG(x|\{X/x\})$ can be calculated according to Eq. (9). $Rel_2$ shows the relevance between features in $X$ and $c$, and $Red_2$ indicates the redundancy in $X$. $F_2$ aims to maximise the relevance $Rel_2$ and minimise the redundancy $Red_2$ among selected features.

3.1.3. *Different weights for relevance and redundancy in GAMI and GAE*

The relevance and redundancy are equally important in Eqs. (10) and (11). In order to investigate the influence of different relative importances for relevance and redundancy, a parameter $\alpha$ is introduced, which is shown by $\alpha_1$ in Eq. (12) and $\alpha_2$ in Eq. (13).

$$F_1 = \alpha_1 * Rel_1 - (1-\alpha_1) * Red_1 \tag{12}$$

$$F_2 = \alpha_2 * Rel_2 - (1-\alpha_2) * Red_2 \tag{13}$$

where $\alpha_1$ and $\alpha_2$ are constant values in $(0, 1)$, which show the relative importance of the relevance. $(1-\alpha_1)$ and $(1-\alpha_2)$ show the relative importance of the reduction of the redundancy. We assume the relevance is more important than the redundancy, so $\alpha_1$ or $\alpha_2$ is set to be larger than $(1-\alpha_1)$ or $(1-\alpha_2)$. When $\alpha_1 = 0.5$ $(1-\alpha_1 = 0.5)$ and $\alpha_2 = 0.5$ $(1-\alpha_2 = 0.5)$, Eqs. (12) and (13) are the same as Eqs. (10) and (11), where the relevance and redundancy are equally important.

### 3.2. *New algorithms*: *NSGAIIMI and NSGAIIE*

GAMI and GAE are single objective algorithms combining the two main objectives of the relevance (indicating the classification performance) and the redundancy (implicitly presenting the number of features). In order to better address feature selection problems, we aim to propose a multi-objective, filter feature selection approach based on evolutionary computation techniques. NSGAII is one of the most

popular evolutionary multi-objective algorithms, proposed by Deb *et al.*[10] The main principle of NSGAII is the use of fast non-dominated sorting technique and the diversity preservation strategy. The fast non-dominated sorting technique is used to rank the parent and child populations to different levels of non-dominated solution fronts. A density estimation based on the crowding distance is adopted to keep the diversity of the population. More details can be seen in the literature.[10]

NSGAII has been successfully used in many areas.[12] However, it has never been applied to filter based feature selection for classification. In this paper, we develop a multi-objective, filter feature selection framework based on NSGAII. Further, two new multi-objective, filter feature selection algorithms, NSGAIIMI and NSGAIIE, are proposed by applying mutual information and entropy as the evaluation criterion in NSGAII.

NSGAIIMI and NSGAIIE aim to minimise the number of features selected and simultaneously maximise the relevance between the feature subset and the class labels. Algorithm 1 shows the pseudo-code of NSGAIIMI and NSGAIIE. After intilisation and the evaluation of individuals, a child population is generated by selection, crossover and mutation operators. Line 8 shows the idea of merging the parent and child populations into a union. Then, the fast non-dominated sorting is performed to identify different levels of Pareto fronts in the union (in Line 10). In this procedure, the non-dominated solutions in the union are called the first non-dominated front, which are then excluded from the union. Then the non-dominated solutions in the new union are called the second non-dominated front. The following levels of non-dominated fronts are identified by repeating this procedure. For the next generation, solutions (individuals) are selected from the top levels of the non-dominated fronts, starting from the first front (from Line 11 to Line 21). When selecting individuals for the new generation, crowding distance is adopted to keep the diversity of the population, which can be seen in Lines 13 and 17. The algorithms repeat the procedures from Line 6 to Line 23 until the predefined maximum generation has been reached.

### 3.3. *New algorithms*: *SPEA2MI and SPEA2E*

In order to further investigate the use of evolutionary multi-objective techniques for filter based feature selection, we propose another multi-objective feature selection framework based on the well-known evolutionary multi-objective algorithm, SPEA2, which has never been applied to filter based feature selection. Further, mutual information and entropy are applied to this framework to propose two new multi-objective algorithms, SPEA2MI and SPEA2E.

SPEA2MI and SPEA2E aim to minimise the number of selected features and simultaneously maximise the relevance between the selected feature subset and the class labels. Algorithm 2 shows the pseudo-code of SPEA2MI and SPEA2E. The main principle of SPEAII is the fine-gained fitness assignment strategy and the use of an archive truncation method. The fine-gained fitness assignment is shown from

**Algorithm 1:** Pseudo-Code of NSGAIIMI and NSGAIIE

```
1  begin
2      Divide Dataset into a Training set and a Test set;
3      Initialise Population based on S (Population size) and D (Dimensionality, number of
       features);
4      Evaluate two objectives of each individual ;          /* number of features and the
       relevance (Rel_1 in NSGAIIMI and Rel_2 in NSGAIIE) on the Training set */
5      Generate Child (new population) by conducting selection, crossover and mutation
       operators;
6      while Maximum Number of Generations is not reached do
7          Evaluate two objectives of each individual in new Child;
8          Merge Child and Population to Union;
9          Empty Population and Child for new generation;
10         Identify different levels of non-dominated fronts F = (F_1, F_2, F_3, ...) in Union ;
           /* Fast non-dominated sorting */
11         while |Population| < S do
12             if |Population| + |F_i| ≤ S then
13                 Calculate crowding distance of each individual in F_i;
14                 Add F_i to Population;
15                 i = i + 1;
16             else
17                 Calculate crowding distance of each particle in F_i;
18                 Sort particles in F_i;
19                 Add the (S − |Population|) least crowded particles to Population;
20             end
21         end
22         Generate Child (new population) by conducting selection, crossover and mutation
           operators;
23     end
24     Calculate the number of features in each solution in F_1;
25     Calculate the classification error rate of the solutions (feature subsets) in F_1 on the
       test set ;                                   /* F_1 is the achieved Pareto front */
26     Return the solutions in F_1;
27     Return the number of features and the test classification error rate of each solution in
       F_1;
28 end
```

Line 8 to Line 10, where the fitness of each individual is the sum of its strength raw fitness and a density estimation. Line 4 shows the intilisation of the archive. The updating process of the archive can be seen from Line 11 to Line 17. When the number of non-dominated solutions is larger than the predefined maximum archive size, the archive truncation method is applied to determine whether a non-dominated solution should be included in the archive or not based on their similarity measured by its distance with its neighbours (Line 16). A new population is constructed by the non-dominated solutions in both the original population and the archive (Line 18). The algorithms repeat the procedures from Line 5 to Line 19 until the predefined maximum generation has been reached.

**Algorithm 2:** Pseudo-Code of SPEA2MI and SPEA2E

```
begin
    Divide Dataset into a Training set and a Test set;
    Initialise the Population based on S (Population size) and D (Dimensionality, number
    of features);
    Create the Archive (empty);
    while Maximum Number of Generations is not reached do
        Evaluate two objectives of each individual ;   /* number of features and the
        relevance (Rel1 in SPEA2MI and Rel2 in SPEA2E) on the Training set */
        Merge Population and Archive to Union;
        Calculate the raw fitness of each individual in Union;
        Calculate the density of each individual in Union;
        Calculate the fitness of each individual in Union ; /* fitness is the sum of the
        raw fitness and the density value */
        Identify the non-dominated solutions in Union and add them to Archive;
        if |Archive| < Maximum Archive Size then
            Add the non-dominated solutions from the remaining Population to Archive ;
            /* Remaining Population excludes the non-dominated solutions that
            have already been added to Archive */
        end
        else if |Archive| > Maximum Archive Size then
            Remove similar solutions to reduce the size of Archive;
        end
        Generate new Population by performing crossover and mutation operators based
        on Archive and Population;
    end
    Calculate the number of features in each solution in Archive;
    Calculate the classification error rate of the solutions in Archive on the test set;
    Return the solutions in Archive;
    Return the number of features and the test classification error rate of each solution in
    Archive;
end
```

## 4. Experimental Design

### 4.1. *Datasets*

Eight datasets (Table 1) are used in the experiments, which were chosen from the UCI machine learning repository.[38] The eight datasets were selected to have different numbers of features, classes and instances and they are used as representative samples of the problems that the proposed algorithms will test on. Since mutual information and entropy are mainly used for discrete variables, all the datasets were selected to have discrete features only, which do not need a discretization process.

In the experiments, all the instances in a dataset are randomly divided into two sets: 70% as the training set and 30% as the test set. The algorithms firstly run on the training set to select feature subsets and then the classification performance (i.e. the classification accuracy or classification error rate) of the selected feature

Table 1. Datasets.

| Dataset | #Features | #Classes | #Instances |
|---|---|---|---|
| Lymphography (Lymph) | 18 | 4 | 148 |
| Mushroom | 22 | 2 | 5644 |
| Spect | 22 | 2 | 267 |
| Leddisplay | 24 | 10 | 1000 |
| Dermatology | 34 | 6 | 366 |
| Soybean Large | 35 | 19 | 683 |
| Chess | 36 | 2 | 3196 |
| Connect4 | 42 | 3 | 67557 |

subsets will be calculated on the test set by a learning algorithm. There are many learning algorithms that can be used here, such as K nearest neighbour, naïve bayes, and DT. As DT is a very commonly used learning algorithm, it is selected in this study to calculate the classification performance of the selected features according to Eq. (14):

$$Error \quad rate = \frac{FP + FN}{TP + TN + FP + FN} \tag{14}$$

where TP, TN, FP and FN stand for true positives, true negatives, false positives and false negatives, respectively.

### 4.2. *Parameter settings*

In the experiments, a library named EvA2[39] is used for the single objective GA and a library named jMetal[40] is used for NSGAII and SPEA2. In all algorithms, the population size is 30 and the maximum generation is 500. A bit-flip mutation operator and single point crossover operator are applied. The mutation rate is $1/n$, where $n$ is the number of available features (dimensionality) and the crossover probability is 0.9. Other parameters are set as the default values in the libraries. For each dataset, all the algorithms have been conducted for 40 independent runs.

For single objective algorithms, GAMI and GAE, a statistical significance test, double tailed Student T-test, is performed between their classification performances and the classification performance obtained by using all features. The significance level in the T-tests was selected as 0.05 (or confidence interval is 95%).

For each dataset, a single solution is obtained by GAMI or GAE in each of the 40 independent runs. Multi-objective algorithms, NSGAIIMI, NSGAIIE, SPEA2MI, or SPEA2E obtain a set of non-dominated solutions in each run. In order to compare these two kinds of results, the 40 solutions that resulted from GAMI and GAE in 40 independent runs are presented individually in the next section. The 40 sets of feature subsets achieved by each multi-objective algorithm are firstly combined into one union set. In the union set, for the feature subsets that contain the same number of features (e.g. $m$), their classification error rates are averaged. The average

classification error rate is assigned as the average classification performance of the subsets with $m$ features. Therefore, a set of average solutions is obtained by using the average classification error rates and the corresponding numbers of features (e.g. $m$). The set of average solutions is called the *average* Pareto front and presented in the next section. Besides the average Pareto front, the non-dominated solutions in the union set are also presented in the next section.

Note that for the same number of features, there are a variety of combinations of features with different classification performance. In different runs, NSGAII may select the same number of features, but with different classification error rates. Therefore, although NSGAIIMI obtained a set of non-dominated solutions, the average solutions in the average Pareto front may dominate each other. This also happens in SPEA2MI, NSGAII-E and SPEA2-E.

### 4.3. *Traditional methods*

In order to examine the performance of the proposed algorithms, two conventional filter feature selection methods (CfsF and CfsB) and a traditional wrapper method (GSBS) are used for comparison purposes in the experiments.

Hall[41] proposed a correlation based filter feature selection method (Cfs) between features and class labels. This method is implemented in Waikato Environment for Knowledge Analysis (Weka)[42] and it needs a search technique. Greedy search in Weka is selected as the search technique to perform both forward and backward selection and they are named as CfsF and CfsB.

The Greedy stepwise based feature selection algorithm is also implemented in Weka. It can move either forward or backward in the search space.[43] We choose a backward search for the greedy stepwise search to conduct a greedy stepwise backward selection (GSBS). GSBS starts with all available features and stops when the deletion of any remaining feature results in a decrease in evaluation, i.e. the classification accuracy.

The three traditional methods produce a unique feature subset, so have a single result for each test set.

## 5. Results and Discussions

This section provides the experimental results and discussions. Tables 2 and 3 show the results of GAMI and GAE with different weights $\alpha$ in the fitness functions. Figures 1 and 2 show the comparisons between the proposed multi-objective algorithms and the single objective algorithms. Table 4 shows the results of three traditional feature selection methods, CfsF, CfsB and GSBS.

### 5.1. *Results of GAMI and GAE*

In Tables 2 and 3, "All" means that all available features are used for classification. "Size" represents the average size of the feature subsets evolved by GAMI and

Table 2. Results of GAMI with different $\alpha_1$ in Eq. (10).

| | Lymph | | | | | | Mushroom | | | | | | Spect | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\alpha_1$ | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 |
| Size | 18 | 14.1 | 9.1 | 6.9 | 5.7 | 4.1 | 22 | 8.1 | 4.1 | 3.6 | 2.9 | 2.0 | 22 | 6.8 | 4.8 | 3.8 | 3.0 | 2.9 |
| Best | 82.22 | 82.2 | 82.2 | 82.2 | 77.8 | 77.8 | 100 | 100 | 99 | 97.9 | 97.9 | 97.8 | 66.3 | 72.5 | 75 | 75 | 71.5 | 71.3 |
| Ave | | 82.2 | 81.1 | 76.8 | 76.7 | 76.9 | | 99.6 | 98.8 | 97.9 | 97.9 | 97.8 | | 68.5 | 69 | 68.8 | 68.6 | 69.0 |
| StdDev | | 0 | 2.6 | 2.3 | 1.9 | 1.8 | | 0.2 | 0.2 | 0.04 | 0.04 | 0.02 | | 2.9 | 3.8 | 4.7 | 4.2 | 4.3 |
| T-test | | = | − | − | − | − | | − | − | − | − | − | | + | + | + | + | + |
| | Leddisplay | | | | | | Dermatology | | | | | | Soybean Large | | | | | |
| $\alpha_1$ | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 |
| Size | 24 | 22.3 | 19.9 | 17.6 | 14.9 | 12.2 | 33 | 30.5 | 17.8 | 11.9 | 8.2 | 6.1 | 35 | 23. | 13.5 | 9.5 | 7.1 | 5.4 |
| Best | 100 | 100 | 100 | 100 | 100 | 100 | 90 | 90 | 90 | 95.5 | 95.5 | 92.7 | 90.7 | 90.7 | 92.2 | 88.8 | 86.8 | 83.4 |
| Ave | | 100 | 100 | 100 | 100 | 100 | | 90 | 89.1 | 90.6 | 90.7 | 85.3 | | 89.5 | 86.9 | 83.8 | 81.6 | 76.8 |
| StdDev | | 0 | 0 | 0 | 0 | 0 | | 0 | 1.0 | 2.7 | 2.5 | 7.5 | | 0.9 | 2.8 | 2.8 | 2.6 | 3.8 |
| T-test | | = | = | = | = | = | | = | − | = | = | − | | − | − | − | − | − |
| | Chess | | | | | | Connect4 | | | | | | | | | | | |
| $\alpha_1$ | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | | | | | | |
| Size | 36 | 14.6 | 11.5 | 9.3 | 7.9 | 6.9 | 42 | 11.9 | 8.2 | 6.9 | 5.4 | 5.4 | | | | | | |
| Best | 98.4 | 97.1 | 95.7 | 95.1 | 95.2 | 95.00 | 74.6 | 70.5 | 69.5 | 67.8 | 68.2 | 68.2 | | | | | | |
| Ave | | 94.9 | 93.7 | 93.0 | 91.5 | 88.9 | | 68.3 | 67.2 | 66.7 | 66.6 | 66.3 | | | | | | |
| StdDev | | 1.4 | 2.6 | 2.1 | 4.7 | 6.4 | | 0.96 | 0.88 | 0.6 | 0.5 | 0.5 | | | | | | |
| T-test | | − | − | − | − | − | | − | − | − | − | − | | | | | | |

Table 3. Results of GAE with different $\alpha_2$ in Eq. (11).

| | Lymph | | | | | | Mushroom | | | | | | Spect | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\alpha_2$ | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 |
| Size | 18 | 9.8 | 9.5 | 8.9 | 7.3 | 5.7 | 22 | 8.9 | 7.6 | 6.8 | 6 | 5.2 | 22 | 18.2 | 16.5 | 14.6 | 11.1 | 8.1 |
| Best | 82.2 | 82.2 | 82.2 | 84.4 | 82.2 | 82.2 | 100 | 100 | 100 | 100 | 100 | 99.5 | 66.3 | 71.3 | 71.3 | 71.3 | 72.5 | 72.5 |
| Ave | | 76.7 | 77.8 | 77.6 | 78.6 | 78.6 | | 99.8 | 99.2 | 98.9 | 98.3 | 97.5 | | 67.6 | 67.6 | 66.8 | 66.4 | 66.2 |
| StdDev | | 2.4 | 2.4 | 3.2 | 3.5 | 2.9 | | 0.2 | 0.6 | 0.7 | 1.0 | 1.3 | | 3.5 | 3.4 | 3.5 | 3.8 | 5.5 |
| T-test | | − | − | − | − | − | | − | − | − | − | − | | + | + | = | = | = |
| | Leddisplay | | | | | | Dermatology | | | | | | Soybean Large | | | | | |
| $\alpha_2$ | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 |
| Size | 24 | 11.7 | 11.2 | 10.9 | 10.1 | 9.3 | 33 | 15.1 | 13.7 | 12.5 | 11.3 | 10.6 | 35 | 21.3 | 20.2 | 19.4 | 17.9 | 17.2 |
| Best | 100 | 100 | 100 | 100 | 100 | 100 | 90 | 95.5 | 95.5 | 95.5 | 94.6 | 94.6 | 90.7 | 92.7 | 91.2 | 90.7 | 92.2 | 91.7 |
| Ave | | 100 | 100 | 99.5 | 99.8 | 99.0 | | 90.0 | 90 | 89.8 | 87.7 | 87.6 | | 84.7 | 85.2 | 84.9 | 83.3 | 82.1 |
| StdDev | | 0 | 0 | 2.3 | 1.4 | 3.7 | | 2.4 | 3.4 | 3.3 | 4.2 | 5.2 | | 4.1 | 3.0 | 3.6 | 4.8 | 4.7 |
| T-test | | = | = | = | = | = | | = | = | = | − | − | | − | − | − | − | − |
| | Chess | | | | | | Connect4 | | | | | | | | | | | |
| $\alpha_2$ | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | All | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | | | | | | |
| Size | 36 | 25.8 | 24.1 | 22.9 | 19.9 | 17.8 | 42 | 34.2 | 34.8 | 34.1 | 31.9 | 31.7 | | | | | | |
| Best | 98.4 | 99.6 | 99.5 | 99.3 | 98.4 | 98.4 | 74.6 | 75.7 | 75.9 | 76.4 | 76.2 | 77.8 | | | | | | |
| Ave | | 98.6 | 98.5 | 98.2 | 97.2 | 96.1 | | 74.5 | 74.4 | 74.2 | 73.7 | 73.8 | | | | | | |
| StdDev | | 0.5 | 0.6 | 0.6 | 1.0 | 1.3 | | 0.5 | 0.8 | 0.9 | 1.0 | 1.4 | | | | | | |
| T-test | | = | = | − | − | − | | = | = | − | − | − | | | | | | |

GAE in the 40 independent runs. "Best", "Ave" and "StdDev" indicate the best, the average and the standard deviation of the 40 test accuracies. "T-test" shows the result of the T-test, where "+" ("−") indicates that the classification performance of GAMI or GAE is significantly better (worse) than that of all features. "=" means they are similar.

According to Table 2, it can be seen that in four of the eight datasets (Lymph, Spect, Leddisplay and Dermatology), GAMI evolved feature subsets that included a smaller number of features and achieved similar or even better classification performance than using all features. In the other four datasets, although the classification performance is slightly worse than using all features, the number of features needed for classification was significantly reduced. For example, in the Mushroom dataset, when $\alpha_1 = 0.9$, the average classification accuracy was only decreased 0.4%, but more than 63% of the features were removed. Moreover, the best classification accuracy achieved by GAMI is the same as using all features.

According to Table 3, in most cases, GAE evolved feature subsets that included a smaller number of features and achieved similar or even better classification performance than using all features. In some cases, the number of features was significantly reduced although the average classification accuracy was slightly decreased. In all datasets, the best classification accuracy evolved by GAE with an appropriate $\alpha_2$ was the same or even better than using all features.

Tables 2 and 3 show that GA with *mutual information* and *entropy* can be successfully applied to feature selection problems. In terms of both the number of features and the classification performance, neither GAMI nor GAE consistently outperformed the other. For both GAMI and GAE, a large $\alpha$ (e.g. 0.9) means the relevance ($Rel_1$ or $Rel_2$) is considered more important than a small $\alpha$ (e.g. 0.5). Therefore, when $\alpha$ is large, GAMI and GAE usually evolved feature subsets with more features and achieved higher classification accuracy than when $\alpha$ is small. While a small $\alpha$ can always reduce the number of features, a large $\alpha$ does not always increase the classification performance. For example, in the Dermatology dataset, the classification performance is the same with different $\alpha$ values, which means that the large feature subsets still have redundancy. For this dataset, $\alpha_1 = 0.6$ seems a generally good value for GAMI, while in GAE, such a value is $\alpha_2 = 0.7$. Therefore, in order to obtain an optimal feature subset, an appropriate weight value $\alpha_1$ or $\alpha_2$ needs to be predefined.

### 5.2. *Results of NSGAIIMI and SPEA2MI*

Figures 1 and 2 show the experimental results of NSGAII and SPEA2 for feature selection with *mutual information* and *entropy* as the evaluation criteria. In order to examine the performance of the multi-objective approaches, their results are compared with that of single objective GA for feature selection. In GAMI and GAE, the number of features is the most important when $\alpha = 0.5$ and the classification performance is the most important when $\alpha = 0.9$. Therefore, the results achieved

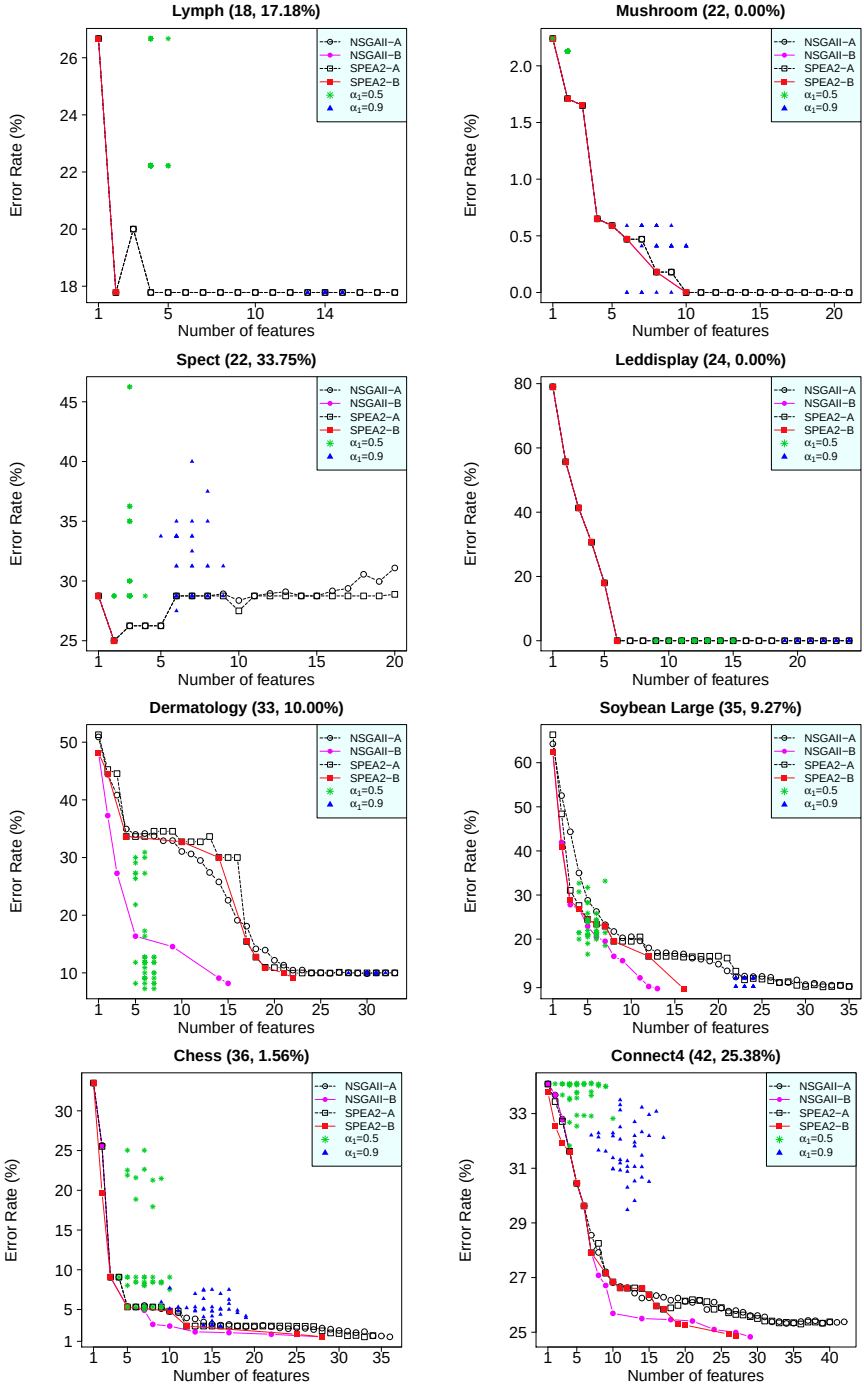


Fig. 1. Experimental Results of GAMI, NSGAIIMI and SPEA2MI.

by GAMI and GAE with $\alpha = 0.9$ and $\alpha = 0.5$ are used as typical results to compare with that of NSGAII and SPEA2.

On the top of each chart, the numbers in the brackets show the number of the available features and the classification error rate using all features. In each chart, the horizontal axis shows the number of features selected and the vertical axis shows the classification error rate. In figures, "-A" stands for the average Pareto front resulting from the 40 independent runs. "-B" represents the non-dominated solutions resulting from the 40 independent runs. $\alpha_1 = 0.5$, $\alpha_1 = 0.9$, $\alpha_2 = 0.5$ and $\alpha_2 = 0.9$ means the 40 solutions of GAMI or GAE with $\alpha_1 = 0.5$, $\alpha_1 = 0.9$, or $\alpha_2 = 0.5$, $\alpha_2 = 0.9$, respectively. Note that in some datasets, GAMI and GAE may evolve the same feature subset in different runs and they are shown in the same point in the chart. Therefore, although 40 results are presented, there may be less than 40 distinct points shown in a chart.

According to Figure 1, in *all* datasets, the average Pareto front of NSGAIIMI, NSGAIIMI-A, contains one or more solutions that selected a smaller number of features and achieved similar or even better classification performance than using all features. In *all* cases, feature subsets in the best Pareto front of NSGAIIMI, NSGAIIMI-B, selected less than half of the available features and achieved similar or better classification performance than using all features. For example, in the Spect dataset, NSGAIIMI-B selected only one feature and improved the classification performance over using all features.

According to Figure 1, SPEA2MI-B in *all* datasets includes one or more feature subsets that selected a small number of features with which DT achieved better classification performance than with all features. In *all* datasets, SPEA2MI-B achieved better classification performance than using all features by selecting only less than half of the available features.

Comparing NSGAIIMI and SPEA2MI with GAMI, it can be seen in most cases, feature subsets in NSGAIIMI-A, NSGAIIMI-B, SPEA2MI-A and SPEA2MI-B outperformed GAMI with $\alpha_1 = 0.5$ and $\alpha_1 = 0.9$ in terms of both the number of features and the classification performance.

Comparing NSGAIIMI with SPEA2MI, in four of the eight datasets (the Lymph, Mushroom, Spect and Leddisplay datasets), NSGAIIMI and SPEA2MI achieved similar or even better results in terms of both the number of features and the classification performance. In the other four datasets, which have more features than the datasets mentioned above, NSGAIIMI achieved slightly better results than SPEA2MI in terms of both the number of features and the classification performance, especially in the Dermatology dataset.

The results in Figure 1 suggest that as multi-objective algorithms, NSGAIIMI and SPEA2MI with *mutual information* as the evaluation criterion can automatically evolve a Pareto front of feature subsets that can reduce the number of features needed for classification and improve the classification performance over using all features.

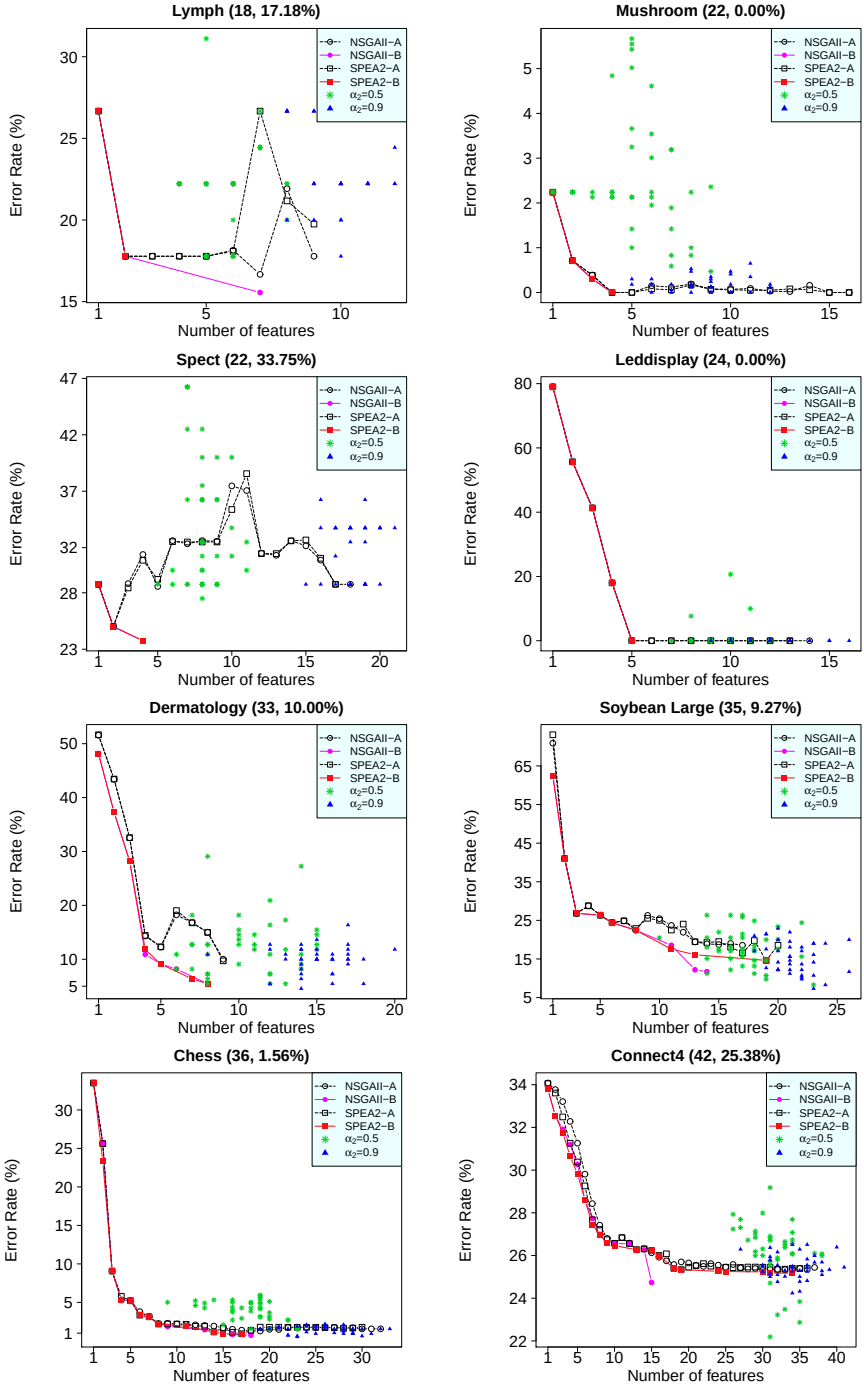


Fig. 2. Experimental Results of GAE, NSGAIIE and SPEA2E.

### 5.3. *Results of NSGAIIE and SPEA2E*

According to Figure 2, in seven of the eight datasets (the exception being the Soybean Large dataset), NSGAIIE-A contains one or more feature subsets that selected a smaller number of features and achieved similar or even better classification performance than using all features. In almost all cases, NSGAIIE-B achieved better classification performance by selecting around one third of the available features. Figure 2 shows that the performance of SPEA2E is similar to that of NSGAIIE in terms of both the classification error rate and the number of features in all datasets.

Comparing NSGAIIE and SPEA2E with GAE, in many cases, the average Pareto fronts, NSGAIIE-A and SPEA2E-A outperformed GAE with $\alpha_2 = 0.5$ in terms of the number of features and the classification performance. In most cases, NSGAIIE-A and SPEA2E-A achieved similar results with GAE with $\alpha_2 = 0.9$, but NSGAIIE-B and SPEA2E-B outperformed GAE.

The results in Figure 2 suggest that NSGAIIE and SPEA2E with *entropy* as the evaluation criterion can automatically evolve a Pareto front of feature subsets that can reduce the number of features needed for classification and improve the classification performance over using all features.

### 5.4. *Mutual information vs. entropy*

Comparing the two evaluation criteria, Figure 1 with Figure 2 show that for the single objective algorithms, GAMI using mutual information usually selected a smaller number of features than GAE using entropy, but GAE achieved slightly better classification performance than GAMI. For the proposed multi-objective algorithms, NSGAIIE and SPEA2E usually evolved a smaller number of features and achieved better classification performance than NSGAIIMI and SPEA2MI. The comparisons suggest that the algorithms with entropy as the evaluation criterion can further increase the classification performance because entropy can discover the multiple-way relevancy and redundancy among a group of features to search for a subset of complementary features. The number of features selected by entropy based algorithms is relatively large because the evaluation is based on a group of features (instead of a pair of features). The number of features in the proposed multi-objective algorithms is always smaller than single objective algorithms, which shows that they can explore the search space more effectively to minimise the number of features. NSGAIIE and SPEA2E can utilise their search ability and the discover multiple-way relevancy to reduce the number of features and simultaneously increase the classification performance.

### 5.5. *Comparisons with traditional methods*

Experimental results of the three traditional feature selection methods, CfsF, CfsB and GSBS, are shown in Table 4.

Comparing Table 4 with Figures 1 and 2, it can be seen that four proposed multi-objective algorithms selected a smaller number of features (excepted for the

Table 4. Results of traditional feature selection methods.

| | Lymph | | | | Mushroom | | | | Spect | | | | Leddisplay | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Method | All | CfsF | CfsB | GSBS | All | CfsF | CfsB | GSBS | All | CfsF | CfsB | GSBS | All | CfsF | CfsB | GSBS |
| Size | 18 | 8 | 8 | 2 | 22 | 3 | 3 | 5 | 22 | 4 | 4 | 6 | 24 | 13 | 13 | 5 |
| Accuracy | 82.2 | 73.3 | 73.3 | 77.8 | 100 | 97.7 | 97.7 | 100 | 66.3 | 70 | 70 | 67.5 | 100 | 100 | 100 | 100 |
| | Dermatology | | | | Soybean Large | | | | Chess | | | | Connect4 | | | |
| Method | All | CfsF | CfsB | GSBS | All | CfsF | CfsB | GSBS | All | CfsF | CfsB | GSBS | All | CfsF | CfsB | GSBS |
| Size | 33 | 17 | 17 | 7 | 35 | 12 | 14 | 12 | 36 | 5 | 5 | 17 | 42 | 6 | 6 | 28 |
| Accuracy | 90 | 87.3 | 87.3 | 90 | 90.7 | 80.5 | 85.4 | 90.2 | 98.4 | 78.1 | 78.1 | 99.1 | 74.6 | 70.3 | 70.3 | 78.8 |

Connect4 dataset) and achieved higher classification performance than the two traditional filter algorithms, CfsF and CfsB, in *all* datasets.

Note that it is not entirely fair to directly compare filter methods with wrapper methods since the wrapper methods use a classifier/learning algorithm within the evaluation process. NSGAIIMI and SPEA2MI as filter algorithms achieved similar or even better results than the wrapper algorithm, GSBS. In six of the eight datasets, NSGAIIE and SPEA2E as filter algorithms achieved better classification performance using fewer features than GSBS. Therefore, in general, the four new multi-objective, filter based algorithms can outperform the traditional wrapper method in terms of both the number of features and the classification performance.

### 5.6. *Further discussion*

#### 5.6.1. *Complexity*

The computational complexities of the algorithms using mutual information and using entropy are considerably different. The running time (evolutionary training time) of the algorithms using joint entropy (GAE, NSGAIIE and SPEA2E) is much longer than those using mutual information (GAMI, NSGAIIMI and SPEA2MI), especially when the dimensionality is large. For example, to finish the 40 runs of experiments on a desktop PC for the Spect dataset, NSGAIIMI took 3.3 seconds while NSGAIIE took around 400 seconds, which is around 120 times longer than NSGAIIMI. For the Chess dataset, NSGAIIMI took 4.5 seconds to finish the 40 runs of the experiments while NSGAIIE took around 9904 seconds, which is around 2184 times longer than NSGAIIMI. This shows that the algorithms using joint entropy did not scale-up well with the dimensionality of the data.

There are two main reasons why NSGAIIE took much longer running time than NSGAIIMI. The first reason is that each calculation of $Rel_1 = \sum_{x \in X} I(x; c)$ (according to Eq. (8)) in NSGAIIMI needs much shorter time than that of $Rel_2 = IG(c|X)$ (according to Eq. (9)) in NSGAIIE. The second reason is that when running the experiments for NSGAIIMI, $I(x; c)$ ($x$ represents a feature and $c$ represents the class label) in $Rel_1$ only has $n$ possible values ($n$ possible combinations of $x$ and $c$), where $n$ is the number of available features. Therefore, the calculation of these $n$ values only needs to be performed once, i.e. at the beginning of the first run of NSGAIIMI on a dataset. During the evolutionary training process, the calculation of $Rel_1$ only needs to refer to the values of $I(x; c)$ and the calculate their sum value. However, for $Rel_2 = IG(c|X)$ ($X$ represents the selected features) in NSGAIIE, during the evolutionary training process, each chromosome has a different $X$. Therefore, each calculation of $Rel_2$ needs to perform Eq. (9), which took longer time than just calculating the sum in $Rel_1$. Although the algorithms using entropy did not scale-up well with the dimensionality of the data, their running time is not very long since they are filter algorithms. For example, for NSGAIIE on the Chess dataset, the average running time for a single run is only around 4 minutes (247 seconds). In our future work, we intend to work on reducing the computa-

tional complexity of the entropy based algorithms, which is out of the scope of this paper.

#### 5.6.2. *Stability*

Experimental results show that the proposed algorithms are quite stable across different independent runs, where the most important feature is always selected by all the algorithms in different runs. In order to show the stability of the proposed algorithms, we take the Spect dataset as an example as the other datasets show a similar pattern.

For the single objective algorithms, GAMI and GAE, a single feature subset was obtained in each run and 40 feature subsets were obtained in the 40 independent runs. Table 5 shows the times of appearance of each feature in the 40 feature subsets (40 runs) evolved by GAMI with $\alpha_1 = 0.5$ or $\alpha_1 = 0.9$ and GAE with $\alpha_2 = 0.5$ or $\alpha_2 = 0.9$. Note that GAMI with $\alpha_1 = 0.5$ usually selected a small number of features (around 3 features, see Table 2), so the corresponding numbers in Table 5 are usually small. GAE with $\alpha_2 = 0.9$ usually selected a relatively large number of features (see Table 3) and the corresponding numbers in Table 5 are usually large. For the multi-objective algorithms, the number of feature subsets reported by each algorithm was 30 and in total, there are 1200 feature subsets obtained by each multi-objective algorithm in the 40 independent runs. Table 6 shows the times of appearance of each feature in the 40 independent runs (1200 feature subsets). In Tables 5 and 6, the three most frequently selected features by each algorithm (the three largest numbers in each row) are highlighted in bold.

For the single objective algorithms, from Table 5, it can be seen that for the same relevance measure, in GAMI with $\alpha_1 = 0.5$ and GAMI with $\alpha_1 = 0.9$, both Features 19 and 22 are the most frequently selected features, which are the same (high) frequencies as Features 1 and 22 in GAE with $\alpha_2 = 0.5$ and with $\alpha_2 = 0.9$. This shows that although different $\alpha_1$ or $\alpha_2$ values lead to different results, Features 19 and 22 or Features 1 and 22 have the largest chances to be selected by GAMI or GAE. Table 5 also show that Feature 22 is one of the top three most frequently selected features in all the four algorithms, which shows that although using different relevance measures and the parameters, GAMI and GAE are reasonably stable algorithms.

For the multi-objective algorithms, as can be seen from Table 6, Features 14, 19 and 22 are the most frequently selected features by NSGAIIMI and SPEA2MI, which are the similar (high) frequencies to Features 17 and 22 in NSGAIIE and SPEA2E. This shows that although they use different search mechanisms, the most frequently selected features in NSGAIIMI and SPEA2MI (NSGAIIE and SPEA2E) are the same or at least similar. Meanwhile, Feature 22 is one of the most frequently selected features in all the four multi-objective algorithms, which shows that the stability of these four multi-objective is reasonably good.

Further comparing Tables 5 and 6, Feature 22 is one of the three most frequently selected features for all of these eight algorithms regardless of the relevance

Table 5. Times of appearance of each feature in the 40 independent runs, where each row shows one method and each column represents one feature.

| Feature ID | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| GAMI ($\alpha_1 = 0.5$) | 0 | 7 | 0 | 0 | 0 | **24** | 0 | 4 | 0 | 0 | 7 | 1 | 0 | 12 | 0 | 0 | 0 | 0 | **34** | 0 | 0 | **28** |
| GAMI ($\alpha_1 = 0.9$) | 34 | 9 | 0 | 10 | 6 | 17 | 4 | 11 | 0 | 3 | 14 | 10 | 17 | **40** | 0 | 3 | 7 | 3 | **40** | 4 | 0 | **40** |
| GAE ($\alpha_2 = 0.5$) | **34** | 4 | 7 | 17 | 11 | 14 | 6 | 6 | 6 | **28** | 15 | 19 | 20 | 9 | 4 | 10 | 18 | 20 | 25 | 15 | 9 | **28** |
| GAE ($\alpha_2 = 0.9$) | **40** | 19 | 19 | 39 | 38 | 11 | 36 | **40** | 28 | 21 | 39 | **40** | **40** | **40** | 28 | 32 | 38 | 33 | 36 | **40** | 29 | **40** |

Table 6. Times of appearance of each feature in the 1200 solutions, where each row shows one method and each column represents one feature.

| Feature ID | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NSGAIIMI | 960 | 440 | 324 | 880 | 150 | 550 | 340 | 639 | 800 | 132 | 490 | 176 | 719 | **1120** | 261 | 185 | 368 | 341 | **1040** | 231 | 122 | **1200** |
| SPEA2MI | 960 | 460 | 298 | 880 | 98 | 560 | 338 | 640 | 800 | 59 | 504 | 179 | 720 | **1120** | 258 | 137 | 419 | 378 | **1040** | 217 | 27 | **1200** |
| NSGAIIE | 557 | 772 | 4 | 832 | 327 | 589 | 334 | 246 | 234 | 777 | 126 | 719 | **853** | 659 | 47 | 28 | **883** | 74 | 151 | 305 | 100 | **1162** |
| SPEA2E | 561 | 824 | 0 | **868** | 335 | 513 | 317 | 239 | 219 | 851 | 83 | 723 | 849 | 669 | 5 | 2 | **891** | 52 | 170 | 319 | 42 | **1126** |

measure, the parameter, the search mechanism, the single objective or the multi-objective algorithms. This shows that the proposed algorithms are stable in that the most important feature is always being selected (assuming Feature 22 is the most important feature). Note that in Table 5, Feature 1 is not selected by GAMI with $\alpha_1 = 0.5$, but was frequently selected by the other three single objective algorithms. The possible reason is feature interaction, which makes Feature 1 become more useful when working together with other features in GAMI with $\alpha_1 = 0.9$, GAE with $\alpha_2 = 0.5$ and GAE with $\alpha_2 = 0.9$, where more features are selected than GAMI with $\alpha_1 = 0.5$.

## 6. Conclusions

This paper aimed to develop an evolutionary multi-objective approach to filter based feature selection with information theory as the evaluation criterion to search for a set of non-dominated feature subsets, which selected a small number of features and achieved similar or even better classification performance than using all features. The goal was successfully achieved by developing four multi-objective feature selection algorithms (NSGAIIMI, SPEA2MI, NSGAIIE, SPEA2E). The four new algorithms were developed by applying two information evaluation criteria (mutual information and entropy) to two multi-objective frameworks. The proposed multi-objective algorithms were examined and compared with single objective GAs based algorithms (GAMI and GAE), and three traditional feature selection methods, CfsF (filter), CfsB (filter) and GSBS (wrapper). In GAMI and GAE, different weights were used in the fitness function to show the relative importance of the classification performance and the number of features.

Experimental results show that with the two filter evaluation criteria, the single objective algorithms, GAMI and GAE, can reduce the number of features in all cases and simultaneously increase the classification performance in some cases. In almost all cases, the proposed multi-objective feature selection algorithms can automatically evolve a set of non-dominated feature subsets that include a smaller number of features and achieve better classification performance than using all features. In most datasets, the proposed four multi-objective algorithms outperformed the single objective algorithms, the two traditional filter feature selection algorithms in terms of both the number of features and the classification performance. With mutual information, NSGAII and SPEA2 can achieve similar or better performance than the wrapper algorithm while with entropy, NSGAII and SPEA2 outperformed the wrapper algorithm in most datasets. NSGAII based approaches achieved similar results to SPEA2 when the number of features is small and slightly better results when the number of features is relatively large.

This work represents the first application of NSGAII and SPEA2 to multi-objective filter based feature selection. Experimental results show that the proposed algorithms can successfully address feature selection problems. It is unfair to directly compare the proposed filter algorithms with wrapper algorithms because

wrappers include a classifier/learning algorithm within the evaluation process. However, the four newly developed multi-objective filter feature selection algorithms outperform the traditional wrapper algorithm, which indicates that the proposed multi-objective algorithms better reflect the nature of feature selection problems and have good potential in this direction.

In the future, we will further investigate multi-objective evolutionary algorithms for feature selection, especially for problems with a large number of features. The claims that filter feature selection methods are more general and less computational expensive than wrappers will also be investigated with the newly developed multi-objective filter based algorithms. We will also work on the application of the proposed algorithms on continuous datasets (not only on discrete datasets) and intend to reduce the complexity of the proposed entropy based algorithms.

## Acknowledgment

This work is supported in part by the National Science Foundation of China (NSFC No. 61170180,61035003), the Key Program of Natural Science Foundation of Jiangsu Province, China (Grant No. BK2011005) and the Marsden Fund of New Zealand (VUW0806) and the University Research Fund of Victoria University of Wellington (200457/3230).

## References

1. M. Dash and H. Liu, Feature selection for classification, *Intelligent Data Analysis* **1**(4) (1997) 131–156.
2. I. Guyon and A. Elisseeff, An introduction to variable and feature selection, *The Journal of Machine Learning Research* **3** (2003) 1157–1182.
3. A. Unler and A. Murat, A discrete particle swarm optimization method for feature selection in binary classification problems, *European Journal of Operational Research* **206**(3) (2010) 528–539.
4. R. Kohavi and G. H. John, Wrappers for feature subset selection, *Artificial Intelligence* **97** (1997) 273–324.
5. A. Whitney, A direct method of nonparametric measurement selection, *IEEE Transactions on Computers* **C20**(9) (1971) 1100–1103.
6. T. Marill and D. Green, On the effectiveness of receptors in recognition systems, *IEEE Transactions on Information Theory* **9**(1) 11–17 (1963).
7. Y. Liu, G. Wang, H. Chen, and H. Dong, An improved particle swarm optimization for feature selection, *Journal of Bionic Engineering* **8**(2) (2011) 191–200.
8. B. Chakraborty, Genetic algorithm with fuzzy fitness function for feature selection, in *IEEE Int. Symp. on Industrial Electronics* (*ISIE'02*), Vol. 1 (2002), pp. 315–319.
9. K. Neshatian and M. Zhang, Genetic programming for feature subset ranking in binary classification problems," in *European Conference on Genetic Programming* (2009), pp. 121–132.
10. K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation* **6**(2) (2002) 182–197.

11. E. Zitzler, M. Laumanns, and L. Thiele, SPEA2: Improving the strength pareto evolutionary algorithm, in *Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems* (2002), pp. 95–100.
12. K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms* (Chichester, UK: John Wiley & Sons, 2001).
13. A. P. Engelbrecht, *Computational Intelligence: An Introduction*, 2nd edn. (Wiley, 2007).
14. J. H. Holland, *Adaption in Natural and Artificial Systems* (University of Michigan Press, 1975).
15. C. Shannon and W. Weaver, *The Mathematical Theory of Communication.* (Urbana: The University of Illinois Press, 1949).
16. L. Cervante, B. Xue, M. Zhang, and 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.
17. K. Kira and L. A. Rendell, "A practical approach to feature selection," *Assorted Conferences and Workshops* (1992), pp. 249–256.
18. C. Cardie, "Using decision trees to improve case-based learning," in *Proc. of the 10th Int. Conf. on Machine Learning* (*ICML*) (1993), pp. 25–32.
19. H. Almuallim and T. G. Dietterich, Learning boolean concepts in the presence of many irrelevant features, *Artificial Intelligence* **69** (1994) 279–305.
20. S. C. Yusta, Different metaheuristic strategies to solve the feature selection problem, *Pattern Recognition Letters* **30** (2009) 525–534.
21. S. Stearns, On selecting features for pattern classifier, in *Proc. of the 3rd Int. Conf. on Pattern Recognition* (Coronado, CA, 1976), pp. 71–75.
22. P. Pudil, J. Novovicova, and J. V. Kittler, Floating search methods in feature selection, *Pattern Recognition Letters* **15**(11) (1994) 1119–1125.
23. C.-L. Huang and C.-J. Wang, A GA-based feature selection and parameters optimizationfor support vector machines, *Expert Systems with Applications* **31**(2) (2006) 231–240.
24. T. M. Hamdani, J.-M. Won, A. M. Alimi, and F. Karray, Multi-objective feature selection with NSGA II, in *8th Int. Conf. on Adaptive and Natural Computing Algorithms* (*ICANNGA'07*), Part I, Vol. 4431 (Springer Berlin Heidelberg, 2007), pp. 240–247.
25. A. J. Soto, R. L. Cecchini, G. E. Vazquez, and I. Ponzoni, Multi-objective feature selection in QSAR using a machine learning approach, *QSAR & Combinatorial Science* **28**(11–12) (2009) 1509–1523.
26. A. Guillén, H. Pomares, J. González, I. Rojas, O. Valenzuela, and B. Prieto, Parallel multiobjective memetic rbfnns design and feature selection for function approximation problems, *Neurocomputing* **72**(16–18) (2009) 3541–3555.
27. B. Huang, B. Buckley, and T.-M. Kechadi, Multi-objective feature selection by using NSGA-II for customer churn prediction in telecommunications, *Expert Systems with Applications* **37**(5) (2010) 3638–3646.
28. Z. X. Zhu, Y. S. Ong, and M. Dash, Wrapper-filter feature selection algorithm using a memetic framework, *IEEE Transactions on Systems, Man,* and *Cybernetics, Part B: Cybernetics* **37**(1) (2007) 70–76.
29. D. Muni, N. Pal, and J. Das, Genetic programming for simultaneous feature selection and classifier design, *IEEE Transactions on Systems, Man,* and *Cybernetics, Part B: Cybernetics* **36**(1) (2006) 106–117.
30. K. Neshatian and M. Zhang, Pareto front feature selection: Using genetic programming to explore feature space, in *Proc. of the 11th Ann. Conf. on Genetic and Evolutionary Computation* (*GECCO'09*) (New York, NY, USA, 2009), pp. 1027–1034.

31. X. Wang, J. Yang, X. Teng, W. Xia, and R. Jensen, Feature selection based on rough sets and particle swarm optimization, *Pattern Recognition Letters*, **28**(4) (2007) 459–471.
32. Z. Pawlak, Rough sets, *Int. Journal of Parallel Programming* **11** (1982) 341–356.
33. M. A. Esseghir, G. Goncalves, and Y. Slimani, Adaptive particle swarm optimizer for feature selection, in *Int. Conf. on Intelligent Data Engineering and Automated Learning* (*IDEAL'10*) (Springer Verlag, Berlin, Heidelberg, 2010), pp. 226–233.
34. S. W. Lin, K. C. Ying, S. C. Chen, and Z. J. Lee, Particle swarm optimization for parameter determination and feature selection of support vector machines, *Expert Systems with Applications* **35**(4) (2008) 1817–1824.
35. H. Ming, A rough set based hybrid method to feature selection, in *Int. Symp. on Knowledge Acquisition and Modeling* (*KAM '08*) (2008), pp. 585–588.
36. R. K. Sivagaminathan and S. Ramakrishnan, A hybrid approach for feature subset selection using neural networks and ant colony optimization, *Expert Systems with Applications* **33**(1) (2007) 49– 60.
37. H. H. Gao, H. H. Yang, and X. Y. Wang, Ant colony optimization based network intrusion feature selection and detection, in *Int. Conf. on Machine Learning and Cybernetics*, Vol. 6 (2005), pp. 49–60.
38. A. Frank and A. Asuncion, UCI machine learning repository, (2010).
39. F. Streichert and H. Ulmer, JavaEvA – A java framework for evolutionary algorithms, Technical Report WSI-2005-06, Centre for Bioinformatics, Tübingen, University of Tübingen, (2005).
40. J. J. Durillo and A. J. Nebro, jMetal: A java framework for multi-objective optimization, *Advances in Engineering Software* **42** (2011) 760–771.
41. M. A. Hall, Correlation-based Feature Subset Selection for Machine Learning, PhD thesis (The University of Waikato, Hamilton, New Zealand, 1999).
42. I. H. Witten and E. Frank, *Data Mining*: *Practical Machine Learning Tools and Techniques*, 2nd edn. (Morgan Kaufmann, 2005).
43. R. Caruana and D. Freitag, Greedy attribute selection, in *Int. Conf. on Machine Learning* (*ICML'94*) (1994), pp. 28–36.