International Journal of Computational Intelligence and Applications
Vol. 13, No. 2 (2014) 1450009 (34 pages)

DOI: 10.1142/S1469026814500096

Imperial College Press
www.icpress.co.uk

BINARY PSO AND ROUGH SET THEORY FOR FEATURE SELECTION: A MULTI-OBJECTIVE FILTER BASED APPROACH

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

*School of Engineering and Computer Science
Victoria University of Wellington, P. O. Box 600
Wellington 6140, New Zealand

†State Key Laboratory of Novel Software Technology
Nanjing University, Nanjing 210046, China
‡Bing.Xue@ecs.vuw.ac.nz

Received 6 October 2013
Revised 1 March 2014
Published 27 June 2014

Feature selection is a multi-objective problem, where the two main objectives are to maximize the classification accuracy and minimize the number of features. However, most of the existing algorithms belong to single objective, wrapper approaches. In this work, we investigate the use of binary particle swarm optimization (BPSO) and probabilistic rough set (PRS) for multi-objective feature selection. We use PRS to propose a new measure for the number of features based on which a new filter based single objective algorithm (PSOPRSE) is developed. Then a new filter-based multi-objective algorithm (MORSE) is proposed, which aims to maximize a measure for the classification performance and minimize the new measure for the number of features. MORSE is examined and compared with PSOPRSE, two existing PSO-based single objective algorithms, two traditional methods, and the only existing BPSO and PRS-based multi-objective algorithm (MORSN). Experiments have been conducted on six commonly used discrete datasets with a relative small number of features and six continuous datasets with a large number of features. The classification performance of the selected feature subsets are evaluated by three classification algorithms (decision trees, Naïve Bayes, and k-nearest neighbors). The results show that the proposed algorithms can automatically select a smaller number of features and achieve similar or better classification performance than using all features. PSOPRSE achieves better performance than the other two PSO-based single objective algorithms and the two traditional methods. MORSN and MORSE outperform all these five single objective algorithms in terms of both the classification performance and the number of features. MORSE achieves better classification performance than MORSN. These filter algorithms are general to the three different classification algorithms.

Keywords: Feature selection; particle swarm optimization; rough set theory; multi-objective optimization.

‡Corresponding author.

1. Introduction

In machine learning and data mining, classification algorithms often suffer from the problem of "the curse of the dimensionality"[1] due to the large number of features in the dataset. Feature selection (or dimension reduction) is proposed as a data preprocessing step to reduce or eliminate irrelevant and redundant features, which aims to reduce the dimensionality, simplify the learnt classifier, reduce the training time, facilitate data visualization and data understanding, and/or increase the classification accuracy.[1,2]

Feature selection is a challenging problem mainly due to two reasons, which are the large search space and feature interaction. For a dataset with m features, the size of the search space is 2^m. Most of the existing algorithms suffer from the problems of being computationally inefficient and becoming stagnated in local optima.[2] Therefore, an efficient global search technique is needed. Evolutionary computation (EC) techniques are argued to be good at global search. One of the relatively recent EC algorithms is particle swarm optimization (PSO).[3,4] Compared with other EC methods, such as genetic programming (GP) and genetic algorithms (GAs), PSO is computationally less expensive, has fewer parameters, and can converge faster.[5] Therefore, researchers recently pay more attention on using PSO to address feature selection tasks.[6,7]

Feature interaction exists in many classification problems. There could be two-way or multi-way interactions among features.[1,8] As a result, a relevant feature may become redundant so that eliminating some of such features will remove or reduce unnecessary complexity. On the other hand, an individually redundant or weakly relevant feature may become highly relevant when working with others. An optimal feature subset is a group of complementary features, but it is difficult to measure the complementary level. Therefore, how to evaluate the goodness (complementary level) of the selected feature subsets is an important issue in feature selection.

Based on the evaluation criteria, feature selection methods are generally classified into two broad classes: wrapper approaches and filter approaches.[1,2] Wrapper approaches include a learning/classification method to evaluate the goodness of the selected feature subsets. Therefore, wrappers often obtain better classification performance than filter approaches, but they suffer from the high computation cost and the loss of generality, i.e., specific to a particular classification algorithm. Filter approaches are independent of any learning algorithm. They are more general and computationally cheaper than wrapper approaches. As a filter feature selection process is independent of any learning algorithm, its performance relies mainly on the goodness of the evaluation criterion. Researchers have introduced different criteria to develop filter approaches, such as consistency measures, information measures and dependency measures.[2,7] However, none of them have become the standard for feature selection. Rough set (RS) theory[9] has been applied to feature selection.[10] However, standard RS has some limitations (details in Sec. 2.3).[11] From a theoretical point of view, Yao and Zhao[11] have shown that probabilistic rough set (PRS) theory

can possibly be a good measure for filter feature selection, but it has seldom been implemented in EC-based filter feature selection approaches.

Most of the existing EC-based feature selection algorithms are single objective, wrapper based methods. However, the use of wrapper algorithms is limited in real-world applications because of being specific to a particular classifier and high computational cost. PSO is computationally cheaper than other EC algorithms, so is a good candidate technique for feature selection. Meanwhile, feature selection is a multi-objective problem with two main conflicting objectives, i.e., maximizing the classification performance and minimizing the number of features selected. Although PSO, multi-objective optimization, or RS has been individually investigated in many works, there are very few studies on using PSO and RS for filter-based multi-objective feature selection. Moreover, due to the constraint that RS only works on discrete data, the datasets used in RS in recent work[10,12–14] only have a small number of features.

1.1. *Goals*

This work aims to present a filter-based multi-objective feature selection approach to obtain a set of nondominated feature subsets. To achieve this goal, we use probabilistic RS to construct two measures: the first measure is to represent the classification performance and the second measure is to represent the number of features. A new single objective method (PSOPRSE) is presented, which combines these two measures into a single fitness function as a direct comparison for the multi-objective approaches. Then two multi-objective methods (MORSN and MORSE) are presented, where MORSN aims to maximize the first measure for the classification performance and minimize the number of features itself, and MORSE aims to optimise the first measure for the classification performance measure and the second measure for the number of features. Furthermore, we will examine and compare the new algorithms with two existing PSO-based single-objective algorithms and two traditional methods on 12 datasets, some of which include several hundreds of features. Specifically, we will investigate:

- whether PSOPRSE can select a small number of features and achieve similar or better classification performance than using all features, and outperform the two existing PSO-based algorithms and the two traditional methods,
- whether MORSN can achieve a set of nondominated feature subsets, and can outperform PSOPRSE,
- whether MORSE can achieve a set of nondominated feature subsets, and can outperform all other methods mentioned above, and
- whether the filter approaches are general to different learning/classification algorithms.

Note that, this work is built on our previous research in Ref. 15 and 16. MORSN was proposed and represents the first PSO and RS-based multi-objective feature selection

algorithm. Due to the page limit, MORSN in Ref. 15 was only tested on six commonly used discrete datasets with a relatively small number of features. MORSN is further tested on six continuous datasets with a large number of features. More importantly, a new RS-based measure (the second measure mentioned above), the new multi-objective algorithm (MORSE) is developed and compared with other methods on 12 datasets in this paper.

1.2. *Organization*

The remainder of the paper is organized as follows. Section 2 presents background information. Section 3 describes the new single objective algorithm and two new multi-objective approaches. Section 4 provides the design of experiments. The results and discussions are given in Secs. 5 and 6 provides conclusions and future work.

2. Background

2.1. *Binary particle swarm optimization*

Particle swarm optimization (PSO)[3,4] simulates the social behaviors of fish schooling and birds flocking. In PSO, each solution of the target problem is represented by a particle. A swarm of particles move ("fly") together in the search space to find the best solutions. For any particle i, a vector $x_i = (x_{i1}, x_{i2}, \ldots, x_{iD})$ is used to represent its position and a vector $v_i = (v_{i1}, v_{i2}, \ldots, v_{iD})$ is used to represent its velocity, where D means the dimensionality of the target problem. During the search process, each particle can remember its best position visited so far called personal best (denoted by *pbest*), and the best previous position visited so far by the whole swarm called global best (denoted by *gbest*). Based on *pbest* and *gbest*, PSO iteratively updates x_i and v_i of each particle to search for the optimal solutions.

Originally, PSO was proposed to address problems/tasks with a continuous search space. To extend PSO to address discrete problems, a binary PSO (BPSO) was developed in Ref. 17, where x_i, *pbest* and *gbest* are limited to 0 or 1. v_i in BPSO represents the probability of an element in the position updating to 1. BPSO updates v_i and x_i of particle i according to Formulae 1 and 2.

$$v_{id}^{t+1} = w \times v_{id}^{t} + c_1 \times r_{i1} \times (p_{id} - x_{id}^{t}) + c_2 \times r_{i2} \times (p_{gd} - x_{id}^{t}), \tag{1}$$

$$x_{id}^{t+1} = \begin{cases} 1, & \text{if } rand() < \dfrac{1}{1 + e^{-v_{id}^{t+1}}}, \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where v_{id}^{t+1} shows the velocity of particle i in the dth dimension in the $t+1$th iteration of the evolutionary process. w is the inertia weight, which indicates the influence of the previous velocity. c_1 and c_2 are acceleration constants. r_{i1}, r_{i2} and $rand()$ are random values, which are uniformly distributed in $[0, 1]$. p_{id} and p_{gd} shows the values of personal best and global best in the dth dimension. A predefined

maximum velocity, $v_{\max}$, is to limit v_{id}^{t+1} to $[-v_{\max}, v_{\max}]$. $rand()$ is randomly chosen from [0,1].

2.2. *Multi-objective optimization*

Multi-objective optimization involves simultaneous optimization of multiple conflicting goals or objectives. The quality of solutions in a multi-objective task are explained by the trade-offs between different conflicting objectives. In mathematical terms, a multi-objective minimization problem can be represented using the following formulae:

$$\text{minimize} \quad F(x) = [f_1(x), f_2(x), \ldots, f_k(x)], \tag{3}$$

subject to:

$$g_i(x) \leq 0, \quad i = 1, 2, \ldots, m, \tag{4}$$

$$h_i(x) < 0, \quad i = 1, 2, \ldots, l, \tag{5}$$

where x shows the decision variables, k is the number of objective functions to be minimized, $f_i(x)$ is one of the objective functions. $g_i(x)$ and $h_i(x)$ are the constraint functions and m and l are integer numbers.

In multi-objective optimization, "Domination" and "Pareto optimum" are two key concepts which consider the trade-offs between objective functions. For example, let a and b be two candidate solutions of the above k-objective minimization task. We can say a is better than b or a dominates b if they meet the following conditions:

$$\forall\, i : f_i(a) \leq f_i(b) \quad \text{and} \quad \exists\, j : f_j(a) < f_j(b), \tag{6}$$

where $i, j \in \{1, 2, 3, \ldots, k\}$.

If no solutions can dominate a, a is a Pareto-optimal/nondominated solution. The Pareto front of the problem is formed by all the Pareto-optimal solutions. A multi-objective algorithm is designed to search for the Pareto front of a multi-objective problem. A feature selection problem can be treated as a two-objective minimization task with the two main objectives of minimizing the number of features and the classification error rate.

2.3. *Probabilistic rough set (PRS) theory*

Rough set (RS) theory[9] is an adaptive mathematical tool to handle uncertainty, imprecision and vagueness. Two of its advantages are that it does not need any prior knowledge about data and all the parameters can be obtained from the given data itself.

In RS, knowledge and information is represented as an information system I. Let U be the universe, which is a finite nonempty set of objects, and A be the features/features that describe each object. $I = (U, A)$. For any $S \subseteq A$ and $X \subseteq U$, an equivalence relation is defined as $IND(S) = \{(x, y) \in U \times U \mid \forall\, a \in S, a(x) = a(y)\}$. If two objects in U satisfy $IND(S)$, they are indiscernible with regards to S. The

equivalence relation, $IND(S)$, induces a partition of U denoted by U/S. U/S further induces a number of equivalence classes. The equivalence class of U/S contains x if $[x]_S = [x]_A = \{y \in U | (x, y) \in IND(S)\}$.

The equivalence classes are regarded as the basic blocks to define rough set approximations. For $X \subset U$, a lower approximation $\underline{S}X$ and an upper approximation $\overline{S}X$ of X with respect to $IND(S)$ are defined as follows[9]:

$$\underline{S}X = \{x \in U \mid [x]_S \subseteq X\} \quad \overline{S}X = \{x \in U \mid [x]_S \cap X \neq \emptyset\}. \tag{7}$$

$\underline{S}X$ includes all the objects that surely belong to the target set X. $\overline{S}X$ contains the objects, which surely or probably belong to the target set X. A rough set is formed by an ordered pair $(\overline{S}X, \underline{S}X)$.

Based on the lower and upper approximations of A, U can be divided into three different regions, which are the positive region $POS_X(S)$, the negative region $NEG_X(S)$ and the boundary region $BND_X(S)$, defined as follows:

$$POS_X(S) = \underline{S}X; \quad NEG_X(S) = U - \overline{S}X; \quad BND_X(S) = \overline{S}X - \underline{S}X. \tag{8}$$

Clearly, the approximation is exact when $BND_X(S)$ is empty.

The reduct is a fundamental concept in RS. A reduct, which is related to a subset of features, is the essential part of an information system. A reduct should achieve similar approximation power of classification to all the original features A. There could be different reducts. Feature selection (or dimension reduction) using RS is usually to remove redundant and irrelevant features to search for the smallest reduct.

$\underline{S}X$ and $\overline{S}X$ in standard RS were defined as two extreme cases.[9] $\underline{S}X$ requires that the equivalence class is a subset of X while $\overline{S}X$ requires that the equivalence class must have a nonempty overlap with X. The degree of their overlap is not taken into account, which will unnecessarily limit its applications. Therefore, researchers investigate probabilistic rough set (PRS) theory to relax the definitions of the lower and upper approximations.[11] The lower approximation is redefined as Eq.(9), where $\mu_S[x]$ is defined as a way to measure the fitness of a given instance $x \in X$.

$$\underline{apr}_S X = \{x \mid \mu_S[x] \geq \alpha\}, \tag{9}$$

where

$$\mu_S[x] = \frac{|[x]_S \cap X|}{|[x]_S|}, \tag{10}$$

α can be adjusted to restrict or relax the lower approximation. An equivalence class includes a number of equivalent objects. If the majority of an object x's equivalent objects in $[x]_S$ are in the target set X, the object x is put in the lower approximation of the target set X. $\underline{apr}_S X = \underline{S}X$ when $\alpha = 1$.

2.4. *Related work on feature selection*

In recent years, researchers have developed different approaches to address feature selection problems.[2,6,8] EC algorithms, such as GAs, GP, PSO and ant colony optimization (ACO) have been used for feature selection. Some typical work in the literature are briefly reviewed in this section.

2.4.1. *Wrapper feature selection approaches*

Sequential forward selection (SFS)[18] and sequential backward selection (SBS)[19] are two typical wrapper feature selection methods. The main difference between SFS and SBS are their starting points. SFS starts with an empty set while SBS starts with all the available features. SFS sequentially selects features until the classification performance is not increased while SBS sequentially remove features until the classification performance is not improved. However, SFS and SBS suffer from the problem of nesting effect. Stearns[20] proposes the "plus-l-take away-r" algorithm to overcome this limitation by performing l times forward selection followed by r times backward elimination. However, it is difficult to find the best values for l and r). To address this challenge, two floating feature selection algorithms are proposed by Pudil *et al.*[21] to automatically determine the values for l and r, which are sequential forward floating selection (SFFS) and sequential backward floating selection (SBFS). However, SFFS and SBFS have the limitation of becoming stagnated in local optima.

EC algorithms have been used to propose wrapper feature selection approaches. Based on a multi-objective GA and neural networks (NN), Oliveira *et al.*[22] propose a modified wrapper feature selection method. Experiments on a handwritten digit recognition dataset show that the proposed algorithm can reduce the number of features and improve the classification performance. However, only one dataset is not sufficient to verify the effectiveness of this method. Zhu *et al.*[23] propose a feature selection method using a memetic algorithm that is a combination of local search and GA. In the proposed algorithm, individual features are first ranked according to a filter measure. GA employs the classification accuracy as the fitness function and deletes or adds a feature according to the ranking information. Experiments show that the proposed algorithm achieves better results than GA and other algorithms. The results also suggest that the performance and the efficiency of the proposed algorithm can be improved by finding a proper balance between genetic search and local search. Neshatian and Zhang[24] propose a feature selection algorithm using GP and naïve bayes (NB), where GP is used to combine features and a set of operators together to find the optimal feature subset. Neshatian *et al.*[25] propose a feature ranking method for feature selection, where each feature is assigned a score according to the frequency of its appearance in a collection of GP trees and the fitness of those trees. Feature selection can be achieved by using the top-ranked features for classification. Experiments show that different classification algorithms can achieve good performance by using only a few top-ranked features. Based on ACO, Kanan and Faez[26] develop a wrapper feature selection algorithm, where both the classification

performance and the number of features are considered. The proposed algorithm outperforms GA and other ACO-based algorithms on a face detection dataset, but its performance has not been tested on other problems.

Marinakis *et al.*[27] propose a wrapper approach based on BPSO and KNN for a real-world medical diagnosis problem, which is called Pap-smear cell classification problem. Results show that the developed method removes around half of the features and achieves good classification performance. Huang and Dun[28] propose a wrapper algorithm for feature selection and parameter optimization in a support vector machine (SVM). In the proposed algorithm, each particle is encoded by two parts, where the first part represents the features in datasets and optimized by binary PSO, and the second part is the parameters in SVM and evaluated by continuous PSO. However, only one dataset with a small number of features is used in the experiments, which cannot demonstrate the full performance of the proposed algorithm. Later, Liu *et al.*[6] also propose a wrapper method to combine feature selection and parameter optimization of SVM in one process. The difference from the method in Ref. 28 is that Liu *et al.*[6] introduce the use of multiple swarms in PSO. Experimental results show that the classification performance of the proposed algorithm is better than that of grid search, standard PSO and GA for feature selection. However, multiple swarms have a larger number of particles and the communication rules between them are complicated, which make the proposed algorithm computationally inefficient. Fdhila *et al.*[29] also apply a multi-swarm PSO algorithm to solve feature selection problems. However, the computational cost of the proposed algorithm is also high because it involves parallel evolutionary processes and multiple subswarms with a relative large number of particles.

To avoid premature convergence, Chuang *et al.*[30] propose a *gbest* resetting strategy in PSO for feature selection, where if the value of *gbest* does not improve over a number of iterations, all its elements will be reset to zero. The proposed algorithm is only compared with one traditional method in terms of the classification performance and no PSO or other EC based algorithms are used for comparisons.

2.4.2. *Filter feature selection approaches*

Hall[31] proposes a filter feature selection method based on the correlation between features and class labels. Almuallim and Dietterich[32] propose a filter algorithm which performs an exhaustive search of all the possible combinations of features, and selects the smallest subset of features. However, performing an exhaustive search is computationally expensive. Relief[33] is a filter algorithm in which each feature has a score indicating its relevance to the class labels. Relief selects all the relevant features. However, the selected feature subset may still have redundancy, because relief does not consider the redundancy between the relevant features.

Based on fuzzy sets, Chakraborty proposes a GA-based filter method[34] and a PSO-based filter method.[35] Comparisons show that the PSO-based algorithm outperforms the GA-based algorithm. Chen *et al.*[13] propose a feature selection algorithm

based on ACO, RS and information theory, where each individual in ACO starts from the core features in RS, and entropy and mutual information are used to guide the search of ACO during the evolutionary training process. Experiments show that the proposed algorithm can be successfully used for feature selection and outperform a GA-based algorithm and a tabu search-based algorithm, but the algorithm has not been tested on datasets with a large number of features. Based on GP, Neshatian and Zhang[36] propose a filter-based multi-objective feature selection algorithm for binary classification problems. They propose a cheap fitness function to improve the computation efficiency and a tree depth control mechanism to allow GP to search space with large subsets if necessary. The GP algorithm can be successfully used for feature selection, but its performance was not compared with any other method.

Based on BPSO, Iswandy and Koenig[37] develop a filter-based algorithm, which employs different weights to linearly combine three objectives, which are evaluated by three filter criteria, into a single fitness function. Results suggest that this algorithm can outperform some other methods on several benchmark problems. Wang *et al.*[10] propose a single objective filter algorithm using an improved BPSO and RS. However, only one classification algorithm is used to evaluate the performance of the selected features, which cannot show the claimed advantage that filter algorithms are more general. In our previous work,[7] two filter-based approaches using PSO and information theory are proposed, where entropy and mutual information are used to evaluate the relevance of the selected features. Results show that the proposed algorithms successfully reduce the number of features for classification and achieve similar or better classification performance than using all features. Bae *et al.*[12] apply a dynamic swarm-based BPSO for feature selection, where RS is used to construct a single objective fitness function. The K-mean algorithm is used to help the proposed algorithm to handle continuous data. Results suggest that this approach can overcome the premature convergence problem and shorten the computation time. However, the number of features in the datasets used in Refs. 7 and 12 is relatively small.

In summary, most of the existing feature selection algorithms are single objective, wrapper approaches, which are computationally more expensive and less general than filter approaches. Meanwhile, the performance of the PRS for feature selection has not been investigated in multi-objective feature selection. Therefore, the development of using PSO and PRS for multi-objective feature selection is still an open issue.

3. Proposed Multi-Objective Approach

Based on PSO and PRS, we propose a new single objective feature selection algorithm (PSOPRSE) and a new multi-objective algorithm (MORSE). To test their performance, two existing single objective feature selection algorithms (PSOPRS and PSOPRSN) and one existing multi-objective algorithm (MORSN) as the baseline are briefly described here which provides some background information for the proposal of the new algorithms.

When using RS for feature selection, a dataset can be regarded as an information system $I = (U, A)$, where all features can be considered as A in the RS. Based on the equivalence relation described by A, U can be partitioned to $U_1, U_2, U_3, \ldots, U_n$, where n is the number of equivalent classes in the dataset. After feature selection, the achieved feature subset can be considered as $P \in A$. Therefore, the fitness of P can be evaluated by how well P represents each target set in U, i.e., a class in the dataset.

3.1. *Existing algorithms: PSOPRS and PSOPRSN*

PSOPRS. As discussed in Sec. 2.3, the definitions of lower approximation and upper approximation limit the application of standard RS. Therefore, a feature selection method (PSOPRS) based on PSO and PRS was proposed in Ref. 14. In PSOPRS, for the target set U_1 in PRS, $\mu_P[x] = \frac{|[x]_P \cap U_1|}{|[x]_P|}$. $\mu_P[x]$ quantifies the proportion of $[x]_P$ is in U_1. $\underline{apr}_P U_1 = \{x \,|\, \mu_P[x] \geq \alpha\}$ defines the lower approximation of P according to U_1. $[x]_P$ does not have to completely contained in U_1. α can be adjusted to restrict or relax $\underline{apr}_P U_1$. Therefore, how well P describes each target in U can be calculated by Eq. 11, which is the fitness function of PSOPRS. Equation (11) essentially measures the number of instances that P correctly makes indistinguishable from others of the same classification.

$$\text{Fitness}_1(P) = \frac{\sum_{i=1}^{n} |\underline{apr}_P U_i|}{|\mathbb{U}|}. \tag{11}$$

PSOPRSN. PSOPRS using PRS can avoid the problems caused by standard rough set, but the number of features is not considered in Eq. (11) in PSOPRS. For the same α value, if there are more than one feature subsets that have the same fitness, PSOPRS does not search for the smaller feature subsets. Therefore, the number of features was added into the fitness function to form another algorithm (PSOPRSN) in Ref. 14. PSOPRSN aims to maximize the ability of the feature subset in separating different classes and also aims to minimize the number of features.

$$\text{Fitness}_2(P) = \gamma * \frac{\sum_{i=1}^{n} |\underline{apr}_P U_i|}{|\mathbb{U}|} + (1 - \gamma) \times \left(1 - \frac{\#\text{features}}{\#\text{totalFeatures}}\right), \tag{12}$$

where $\gamma \in (0, 1]$ represents the relative importance of the feature subset's ability in terms of separating different classes. $(1 - \gamma)$ represents the importance of the number of features. When $\gamma = 1.0$, PSOPRSN is the same as PSOPRS.

3.2. *New single objective algorithm: PSOPRSE*

In PSOPRSN, the number of features is directly considered in the fitness function. By adjusting the value of γ, PSOPRSN is expected to find a smaller feature subset and maintain or slightly reduce the classification performance. However, this might be not achieved by PSOPRSN because of the nature of probabilistic rough set. In RS, patterns in the datasets are extracted by the equivalence classes because they are

used to determine the member of the lower and upper approximations of a class (target set). A small number of features can describe a large number of equivalence classes. For example, 12 binary features can define at most 1048576 (2^{12}) equivalence classes. However, there can be several thousands of small equivalence classes only including very few instances. If one equivalence class contains a slightly larger number of instances, it will dominate others. As a result, the obtained reduct will only have the information that can identify this particular class. Therefore, PSOPRSN can obtain a small reduct, but may potentially lose the generality and can not perform well one unseen test data.

In order to solve the problem, we consider the size of the equivalence classes and propose a new measure to minimize the number of features in the reduct, $\frac{\sum_{x\in\{\text{the equivalence classes}\}}\frac{|x|}{|\mathbb{U}|}}{\#\text{ of equivalence classes}}$, which aims to minimize the number of equivalence classes and maximize the number of instances in each equivalence class. Based on this new measure, we propose a new PSO-based single objective algorithm (PSOPRSE), where Eq. (13) is used as the fitness function.

$$\text{Fitness}_3(P) = \frac{\sum_{x=1}^{n}|\underline{apr}_P X_i|}{|\mathbb{U}|} + \frac{\sum_{x\in\{\text{the equivalence classes}\}}\frac{|x|}{|\mathbb{U}|}}{\#\text{ of equivalence classes}}. \tag{13}$$

Note that, the two parts in Eq. (13) are related to each other and both of them have the range of (0,1]. We treat them equally important and do not use any weighting factor.

3.3. *New multi-objective algorithm: MORSE*

PSOPRSN and PSOPRSE combine the two main objectives of feature selection into a single fitness function. However, γ in PSOPRSN needs to be predefined and its best value is problem-dependent. Therefore, a PSO and PRS-based multi-objective feature selection algorithm is needed. However, PSO was originally proposed for single objective optimization. Based on the ideas of mutation, crowding and dominance, Sierra and Coello[38] proposed a multi-objective PSO approach, which is a continuous algorithm and has achieved good performance. Since feature selection is a binary problem, we extended it to a binary version of multi-objective PSO. We proposed a multi-objective feature selection method (MORSN),[15] which is based on PRS and the extended binary multi-objective PSO. The two objectives in MORSN are to maximize the ability of the feature subset to separate different classes of instances, which is evaluated by $\frac{\sum_{i=1}^{n}|\underline{apr}_P U_i|}{|\mathbb{U}|}$, and to minimize the number of features, which are the two parts in Eq. (12) without using the predefined γ.

As discussed in Sec. 3.2, the number of features as a measure in the fitness function might not work well in the situation of using PRS for feature selection. We propose another multi-objective feature selection algorithm (MORSE) based on PSO and PRS, where the two objectives are to maximize the ability of the feature subset

to separate different classes of instances, evaluated by $\frac{\sum_{i=1}^{n}|\underline{apr}_P U_i|}{|\mathbb{U}|}$, and to minimize the number of equivalence classes, which are the two parts in Eq. (13) without using the predefined γ.

Algorithm 1 shows the pseudo-code of MORSN and MORSE. A leader set, a crowding factor, a binary tournament selection, two mutation operators, and parameters determination are important mechanisms employed by MORSN and MORSE to improve their performance. The use of the leader set, the crowding factor and the binary tournament selection is to address the key problem in extending single PSO to multi-objective PSO, which is how to determine the global best, *gbest*, for each particle. The leader set is used by MORSN and MORSE to store the

Algorithm 1: Pseudo-Code of MORSN and MORSE

```
begin
    split the instances to a Training set and a Test set;
    initialise the position, x, and the velocity, v, of each particle;
    split the swarm into three sub-swarms;
    initialise the leader set, LeaderSet
    calculate the crowding distance of each solution in LeaderSet;
    while Maximum Iteration has not been reached do
        for i=1 to Swarm Size do
            use the binary tournament selection to choose two solutions from
            LeaderSet;
            set the less crowded solution as the gbest for particle i;
            calculate v_i according to Equation 1;
            calculate x_i according to Equation 2;
            apply mutation operators if applied;
            evaluate two objective values of each particle; /* the relevance
            (evaluated by Σ_{i=1}^{n} |apr_P U_i| / |U|) and the number of features
            in MORSN or the number of equivalent classes in MORSE
            */
            update the pbest of particle i;
        end
        identify the non-dominated solutions (feature subsets) and update
        LeaderSet;
        evaluate the crowding distance of each member in LeaderSet;
    end
    calculate the classification error rate of the final solutions on the test set;
    return the obtained solutions, i.e., the selected individual features, the
    number of selected features and the training and testing classification
    error rates;
end
```

nondominated feature subsets (solutions) obtained by the whole swarm from where the *gbest* of each particle is chosen. As the number of nondominated feature subsets may increase fast, the maximum size of the leader set is defined as the total number of particles in the swarm. When a larger number of nondominated solutions are obtained, a crowding factor is applied to determine which of them should be put and kept in the leader set over future iterations. When selecting a *gbest*, MORSN and MORSE employs the binary tournament selection to choose two feature subsets from the leader set and the less crowded one will be selected as the *gbest*.

In order to avoid the loss of diversity of the population, MORSN and MORSE use two different mutation operators, which are uniform mutation and nonuniform mutation. In uniform mutation, a decision variable has a fixed range of variability over iterations while in nonuniform mutation, the variability range becomes smaller and smaller over iterations. Uniform mutation facilitates global search (exploring) capability while nonuniform mutation facilitates local search (exploiting) ability. The use of both mutation operators can improve the search ability of the algorithms by balancing the local and global search abilities. To apply these two operators, when initializing the population, particles in MORSN and MORSE are divided to three sub-swarms. The uniform mutation operator is applied to the first sub-swarm with an attempt to ensure the global search capability to quickly explore the search space. The nonuniform mutation operator is applied to the second sub-swarm to maintain the local search capability to exploit better solutions. The situation of not using any mutation operator is also considered and applied to the third sub-swarm. These three sub-swarms are not independent to each other. They share one leader set to choose *gbest* for each particle in the sub-swarm. This allows them to communicate with each other and share the success of different behaviors.

In MORSN and MORSE, instead of using fixed values, w, c_1 and c_2 are set as random numbers in different ranges. w is randomly chosen from $[0.1, 0.5]$, and c_1 and c_2 are randomly selected from $[1.5, 2.0]$. This is employed as a convenient way to handle the parameter tuning issue for test problems of varying difficulty.

4. Design of Experiments

In order to examine the performance of the proposed algorithms, we first choose six discrete datasets (listed in Table 1) from UCI machine learning repository[39] in the

Table 1. Datasets.

Dataset	# Features	# Classes	# Instances
Lymphography (Lymph)	18	4	148
Spect	22	2	267
Dermatology	33	6	366
Soybean Large	35	19	307
Chess	36	2	3196
Statlog	36	6	6435

experiments. All the six datasets are categorical data because rough set theory only works on discrete values. They have different number of instances, features and classes, which are used as representative examples of the problems that the proposed algorithms will address.

In each dataset, 70% of the instances are chosen as the training set and the other 30% are the test set. The filter algorithms first run on the training set in order to select a feature subset(s). The performance of the selected feature subset(s) is then evaluated by a learning/classification algorithm on the unseen test set. Note that, as filter approaches, the feature selection (evolutionary training) process of the proposed algorithms is independent of the learning algorithm and they only run on the test set to evaluate the classification performance of the obtained subsets of features. Almost all learning algorithms can be used here. In order to investigate whether filter feature selection methods are general, three commonly used learning algorithms, decision trees (DT), Naïve Bayes (NB) and K-nearest neighbor algorithms with $K = 5$ (5NN), are used in the experiments.

All the α values should be larger than 0.5, because the lower approximation in probabilistic rough set theory should have the majority (at least have half) of the instances that belong to the target set. Based on our preliminary work,[14] $\alpha = 0.8$ is chosen in the experiments for all methods.

In all the algorithms, each particle is represented by a binary string, whose length is the total number of features in the dataset, which also represents the dimension of the solution space. "1" in the binary string indicates that the corresponding feature is selected and "0" indicates that this feature is removed. The fully connected topology is used in BPSO, the population size is 30 and the maximum iteration is 200 in all the algorithms. In the three single objective algorithms, PSOPRS, PSOPRSN and PSOPRSE, the swarm size is 30, the fully connected topology is used in PSO. $w = 0.7298$, $v_{\max} = 6.0$, $c_1 = c_2 = 1.49618$.[4] In PSOPRSN, two different γ values (0.9 and 0.5) are used to represent the different relative importance of the classification performance and the number of features in the fitness function is 12. In the two multi-objective algorithms, MORSN and MORSE, w is randomly chosen from [0.1,0.5], and c_1 and c_2 are randomly selected from [1.5, 2.0]. The mutation rate is $1/n$, where n is the total number of features in the dataset. These values are based on the settings of an equivalent algorithm in the literature.[38] Each algorithm has been conducted for 50 independent runs on each dataset.

Wilcoxon test is performed with the significance level of 0.05 to test the results of PSOPRSE with that of using all features, PSOPRS, and PSOPRSN. The Wilcoxon test may need to be performed twice to compare the classification performance of two methods, e.g., PSOPRSE and PSOPRS. In the first test, the null hypothesis is that the classification performance of the two methods are similar to each other. If the p-value is equal or larger than 0.05, the null hypothesis is true, i.e., there is no significant difference between the classification performance of the two methods. If the p-value is smaller than 0.05, the null hypothesis is not true and the second test needs to perform. The null hypothesis in the second test is that the accuracies of

PSOPRSE are significantly higher than PSOPRS. If the p-value is smaller than 0.05, the accuracies of PSOPRSE are significantly lower than PSOPRS. Otherwise, the accuracies of PSOPRSE are significantly higher than PSOPRS.

Two traditional filter feature selection algorithms (CfsF and CfsB) in Weka[40] are used for comparison purposes in the experiments and the classification performance is calculated by DT. In order to further investigate the performance of the proposed algorithms, we will also use six continuous datasets[39] with a large number of features in the experiments, which are listed in Sec. 6. The six continuous datasets are discretized using Weka and used in the experiments. The proposed approaches are examined and compared with other methods on these datasets.

5. Results and Discussions

In this section, first, the results of three single objective algorithms, PSOPRS, PSOPRSN and PSOPRSE are discussed. Second, we compare the performance of PSOPRSN with that of MORSN. Third, we discuss the results of PSOPRSE and MORSE. Fourth, we compare the two multi-objective algorithm, MORSN with MORSE. Finally, the results of these five algorithms are compared with two traditional filter feature selection algorithms, CfsF and CfsB. The five algorithms are further tested on the discretized continuous datasets with a large number of features.

5.1. *PSOPRSE, PSOPRS and PSOPRSN*

Table 2 shows the results of PSOPRS and PSOPRSN with $\gamma = 0.9$ and $\gamma = 0.5$, and PSOPRSE. The classification performance (error rates) of the selected feature subsets were evaluated by DT, NB and 5NN on the test set of each dataset. In Table 2, "All" means that all of the available features are used for classification. "Size" means the average number of features selected in the 50 independent runs. "Mean", "Best" and "StdDev" represent the mean, the best and the standard deviation of the classification error rate of the 50 feature subsets obtained by each algorithm in the 50 independent runs. The results of the Wilcoxon tests are shown in the last column, where "+" or "−" means the classification performance of PSOPRSE is significantly better or worse than that of "All", PSOPRS, or PSOPRSN. "=" means there is no significant difference between their classification performance.

PSOPRS. According to Table 2, in almost all cases, PSOPRS selected around two thirds of the available features. By using the selected features, DT, NB and 5NN achieved similar or better classification performance than using all the available features. Although in some cases, the average classification error rate of the obtained feature subsets is slightly higher than that of all features, the best classification performance is better than that of all features in almost all cases. The results suggest that PSOPRS based on BPSO and probabilistic rough set theory can successfully reduce the number of features needed for classification.

Table 2. Results of PSOPRS, PSOPRSN ($\gamma = 0.9$ and $\gamma = 0.5$) and PSOPRSE.

			DT			NB			NN		
Dataset	Method	Size	Best	Mean ± StdDev	Test	Best	Mean ± StdDev	Test	Best	Mean ± StdDev	Test
Chess	All	36	1.5		−	12.11		=	6.76		+
	PSOPRS	29.97	1.5	1.72 ± 27.9E-2	+	8.73	11.32 ± 1.4E0	=	4.98	6.02 ± 62.5E-2	+
	$\gamma = 0.9$	14.43	1.31	2.15 ± 31.8E-2	+	4.98	7.96 ± 2.07E0	−	4.88	10.2 ± 4.41E0	+
	$\gamma = 0.5$	5.4	5.92	6.39 ± 82.1E-2	+	5.92	6.71 ± 1.11E0	−	7.23	28.18 ± 17E0	+
	PSOPRSE	29.27	1.31	1.57 ± 20.8E-2		8.92	11.46 ± 1.59E0		3.94	5.11 ± 67.7E-2	
Dermatology	All	34	17.21		+	4.1		−	4.92		−
	PSOPRS	21	2.46	13.99 ± 4.76E0	+	1.64	6.53 ± 3.23E0	=	2.46	8.11 ± 3.25E0	=
	$\gamma = 0.9$	8.13	8.2	25.19 ± 6.92E0	+	8.2	19.56 ± 5.73E0	+	12.3	21.83 ± 5.59E0	+
	$\gamma = 0.5$	6.8	8.2	27.13 ± 9.48E0	+	8.2	22.65 ± 6.68E0	+	11.48	26.67 ± 5.78E0	+
	PSOPRSE	11.47	3.28	8.06 ± 3.1E0		4.1	7.6 ± 2.14E0		2.46	7.16 ± 4.05E0	
Lymph	All	18	24.49		−	12.24		−	18.37		−
	PSOPRS	11.77	20.41	27.69 ± 6.82E0	=	8.16	15.44 ± 3.52E0	−	14.29	22.45 ± 4.65E0	=
	$\gamma = 0.9$	5.03	32.65	33.27 ± 3.3E0	=	20.41	22.38 ± 36.6E-2	+	24.49	24.69 ± 1.1E0	+
	$\gamma = 0.5$	5	32.65	32.65 ± 31E-4	=	22.45	22.45 ± 10E-4	+	24.49	24.49 ± 2E-4	+
	PSOPRSE	6.53	24.49	31.36 ± 9.84E0		14.29	18.1 ± 1.56E0		22.45	23.27 ± 2.33E0	
Soybean	All	35	18.06		=	9.69		−	9.25		−
	PSOPRS	21.67	12.78	19.47 ± 4.33E0	=	10.13	15.04 ± 2.85E0	−	13.22	19.3 ± 4.22E0	−
	$\gamma = 0.9$	9.7	20.26	27.64 ± 2.87E0	+	16.74	24.2 ± 4.63E0	+	24.67	31.56 ± 3.75E0	+
	$\gamma = 0.5$	7.67	17.62	27.93 ± 4.24E0	+	17.62	25.14 ± 3.93E0	+	25.11	33.05 ± 4.14E0	+
	PSOPRSE	19.17	14.98	19.09 ± 2.94E0		14.54	18.66 ± 2.48E0		20.7	26.17 ± 2.91E0	
Spect	All	22	19.1		−	23.6		−	17.98		−
	PSOPRS	17.5	17.98	19.96 ± 1.96E0	−	17.98	22.96 ± 1.55E0	−	15.73	18.76 ± 1.57E0	−
	$\gamma = 0.9$	14	17.98	17.98 ± 25E-4	−	20.22	20.86 ± 55.7E-2	−	16.85	16.85 ± 39E-4	−
	$\gamma = 0.5$	3.1	17.98	25.32 ± 1.53E0	−	15.73	24.49 ± 3.34E0	−	15.73	15.73 ± 3E-4	−
	PSOPRSE	16.2	29.21	34.53 ± 3.29E0		28.09	29.21 ± 1E0		38.2	39.33 ± 1.74E0	

Table 2. (*Continued*)

			DT			NB			NN		
Dataset	Method	Size	Best	Mean ± StdDev	Test	Best	Mean ± StdDev	Test	Best	Mean ± StdDev	Test
Waveform	All	40	25.21		=	20.29		−	20.95		−
	PSOPRS	24.47	22.81	25.21 ± 1.88E0	−	18.73	22.28 ± 1.99E0	−	19.75	24.77 ± 2.64E0	=
	$\gamma = 0.9$	8.03	25.33	31.4 ± 5.62E0	+	23.17	30.04 ± 5.35E0	+	25.21	34.57 ± 6.41E0	+
	$\gamma = 0.5$	7	25.63	29.79 ± 2.24E0	+	25.63	29.1 ± 2.11E0	+	28.99	33.27 ± 2.78E0	+
	PSOPRSE	19.87	22.81	27.27 ± 4.71E0		18.73	24.79 ± 5.17E0		18.97	27.79 ± 6.59E0	
Statlog	All	36	13.61		−	17.39		−	9.93		−
	PSOPRS	25.63	12.87	14.4 ± 65.9E-2	=	17.16	17.87 ± 37.9E-2	=	9.79	10.68 ± 51.6E-2	−
	$\gamma = 0.9$	13.27	13.85	15.36 ± 61E-2	+	17.3	18.62 ± 57.1E-2	+	10.26	11.8 ± 61.8E-2	+
	$\gamma = 0.5$	7.57	14.22	16.31 ± 1.03E0	+	18.41	20.05 ± 99.9E-2	+	11.84	13.86 ± 1.06E0	+
	PSOPRSE	20.1	13.38	14.62 ± 77.4E-2		16.88	17.97 ± 50.3E-2		10.26	11.03 ± 45.4E-2	

PSOPRSN. According to Table 2, by considering the number of features in the fitness function, PSOPRSN further reduced the number of features selected. PSOPRSN with a small γ selected a smaller number of features than with a relatively large γ. The reason is that a smaller γ means the number of features in PSOPRSN is more important than a relatively large γ. However, when the number of features decreases, the classification performance also decreases. When $\gamma = 0.5$, PSOPRSN could not improve the classification performance on any of the three learning algorithms. This is consistent with our hypothesis discussed in Sec. 3.2. Without considering the size of the equivalence class, PSOPRSN could reduce the number of features in the reduct, but also reduce the generality of the reduct. Meanwhile, the value of γ needs to be predefined. A larger γ was supposed to represent that the classification performance is more important than a smaller γ, but the results in Table 2 show that the classification performance of $\gamma = 0.9$ is not always better than that of $\gamma = 0.5$, such as in the waveform dataset. The reason might be that the PSOPRSN with $\gamma = 0.5$ further remove some redundant features, which also reduce the complexity of the classification algorithms. This suggests that the parameter γ, which is to balance the relative importance of the classification performance and the number of features, is difficult to determine in advance. It also indicates that it is necessary to develop a multi-objective algorithm to solve feature selection problems.

PSOPRSE. From Table 2, we can observe that in almost all cases, PSOPRSE selected half or less than half of the available features and improved the classification performance over using all the available features. Although the average classification performance of the selected features is slightly worse than that of all features in some cases, their best classification performance is superior to that of all features in almost all cases. The results suggest that PSOPRSE considering both the classification power of the selected features and the number of equivalence classes can successfully select a smaller number of relevant features and achieve similar or improve the classification performance of all features.

Comparisons Between PSOPRS, PSOPRSN and PSOPRSE. The results show that PSOPRSE outperformed PSOPRS in terms of both the number of features and the classification performance in most cases. For example, in the Dermatology dataset using DT as the classification algorithm, PSOPRS selected around 21 features from the 34 available features and obtained a classification error rate of 13.99% and PSOPRSE further reduced the average number of features to 9.87 and reduced the classification error rate to 7.92%. This suggest that, by considering the number of equivalence classes in the fitness function, PSOPRSE can further reduce/remove some redundant or irrelevant features but keep the classification power of the remaining features to maintain or even increase the classification performance of PSOPRS.

Both PSOPRSN and PSOPRSE consider the classification power of the features represented by $\frac{\sum_{i=1}^{n} |\underline{apr}_P U_i|}{|\mathbb{U}|}$ and the number of features, which is represented by the

number of features in PSOPRSN and by the number of equivalence classes in PSOPRSE. Compared with PSOPRSN, one advantage of PSOPRSE is that PSOPRSE does not need to predefine the parameter γ. PSOPRSN achieved a smaller number of features, for all the three learning algorithms, but the classification performance in PSOPRSN is much worse than in PSOPRSE in most cases. The main reason is that without considering the size of the equivalence classes, PSOPRSN obtained a small number of features, but it lost the generality and could not achieve good performance on unseen test data. Since the classification performance is usually more important than the number of features in feature selection problems, PSOPRSE can be regarded as a better feature selection approach than PSOPRSN.

Generally, PSOPRS, PSOPRSN and PSOPRSE based on PSO and probabilistic rough set theory can be successfully used for feature selection. PSOPRSE that uses the number of equivalence classes to represent the number of features can achieve better performance than PSOPRS and PSOPRSN. However, it is unknown whether more features can be removed and the classification performance can still be maintained or even increased. Meanwhile, as shown by PSOPRSN, the parameter to balance the relative importance of the number of features and the classification performance is difficult to define in advance. Therefore, it is needed to treat a feature selection problem as a multi-objective task.

5.2. *Results of MORSN*

In the experiments, single objective algorithms (PSOPRSN and PSOPRSE), only obtained a single feature subset/solution in each independent run (50 solutions in the 50 runs). Multi-objective algorithms (MORSN and MORSE) achieved a set of non-dominated solutions in each independent run. To compare the performance of PSOPRSN with MORSN, the 50 feature subsets resulted from PSOPRSN are presented. The 50 sets of solutions obtained by MORSN are stored in a union. The classification performance of the feature subsets, which have the same number (e.g., c) of features, are averaged. A new set of average solutions named the *average* Pareto front are obtained, where each single solution is constructed by assigning the average classification performance to the corresponding number c. Meanwhile, the non-dominated solutions in the union set are called the *best* Pareto front. Both the *average* and *best* Pareto fronts are presented here and compared with the solutions obtained by PSOPRSN.

Figure 1 shows the results of MORSN, PSOPRSN with $\gamma = 0.5$ and $\gamma = 0.9$ on the test sets, where DT was used as the classification algorithm. In each figure, each chart shows the solutions of one dataset used in the experiments, the horizontal axis and the vertical axis show the number of features and the classification error rate, respectively. The total number of features and the classification error rate using all the available features are shown in the brackets on the top of each chart. The results of using 5NN or NB as the classification algorithm show a similar pattern to that of using DT and the detailed results are not presented here to save space. All the

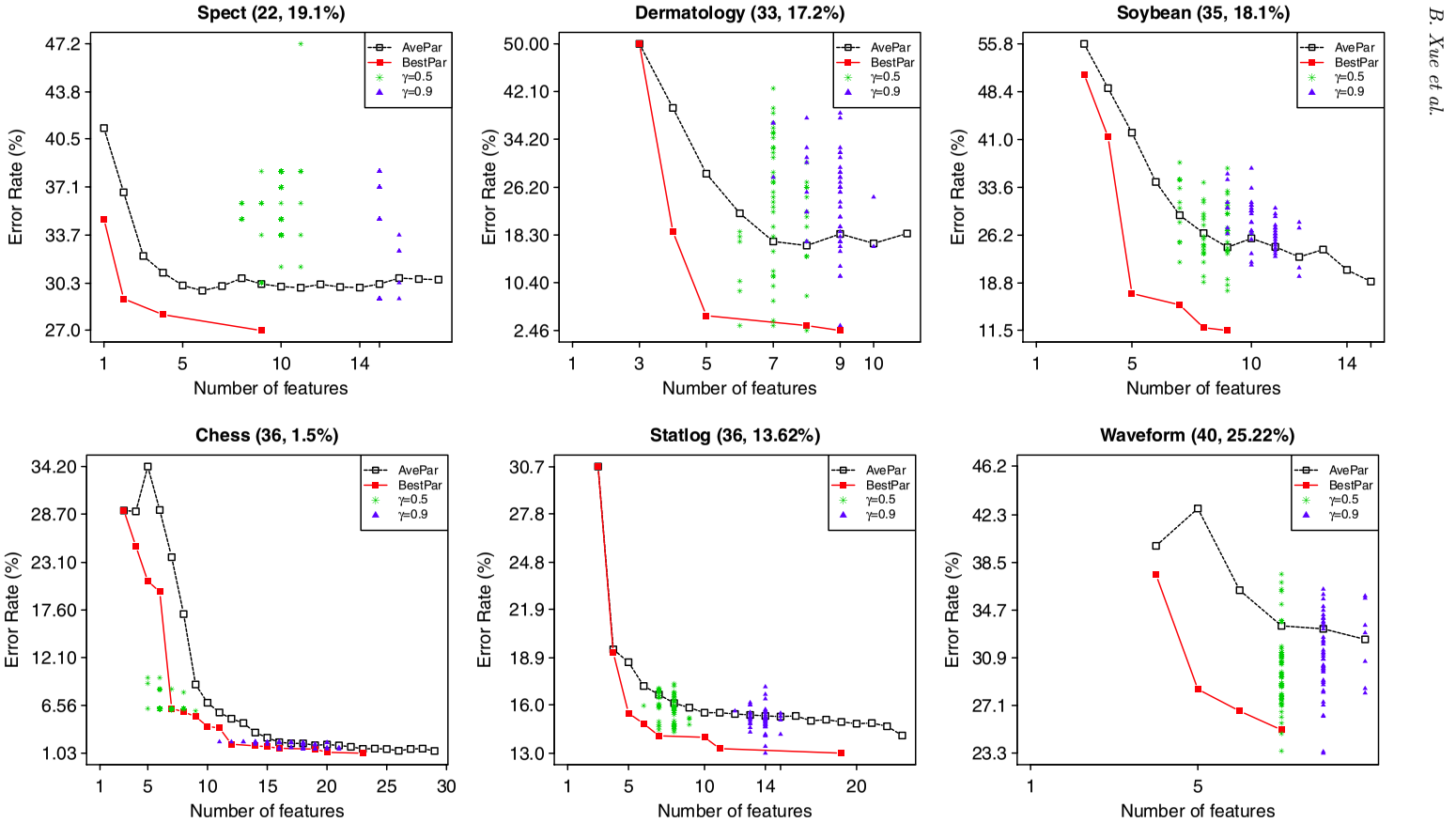

Fig. 1. Results of MORSN and PSOPRSN on test sets evaluated by DT.

detailed results of using 5NN and NB in this paper can be seen from https://ecs.victoria.ac.nz/Groups/ECRG/OnlineSupplimentaryMaterials.

In Fig. 1, "MORSN-AvePar" and "MORSN-BestPar" stand for the *average* and the *best* Pareto fronts resulted from MORSN over the 50 independent runs. $\gamma = 0.5$ and $\gamma = 0.9$ show the results of PSOPRSN with $\gamma = 0.5$ and $\gamma = 0.9$, respectively. In some datasets, the feature subsets evolved by PSOPRSN in different runs may have the same number of features and same classification performance, which are plotted at the same point in the figure. Therefore, although all the 50 solutions are plotted for $\gamma = 0.5$ ($\gamma = 0.9$), some charts may have fewer than 50 distinct points.

MORSN Using DT. According to Fig. 1, in most cases, the average Pareto front of MORSN (MORSN-AvePar) contains two or more solutions, which included a smaller number of features and obtained a similar or lower classification error rate than using all the available features. Note that, for a certain number (e.g., c), there are a variety of combinations of c features, but they achieved different classification performance. In different runs, MORSN may obtain a number of feature subsets all of which includes c features, but different classification error rates. After averaging their classification performance, the solution with c features in the average Pareto front may have worse (better) classification performance than with $c-1$ ($c+1$) features. Therefore, some solutions in the average Pareto front may be dominated by some others, although the feature subsets achieved in each run are nondominated to each other. This also happens when using 5NN or NB as the classification algorithms and in the results of MORSE in Sec. 5.3.

According to Fig. 1, in *all* datasets, the nondominated solutions of MORSN-BestPar selected one or more feature subsets, which included less than one third of the features and reduced the classification error rate of using all features.

Comparisons Between MORSN and PSOPRSN Using DT. In most datasets, solutions in AvePar in MORSN achieved similar results to both $\gamma = 0.5$ and $\gamma = 0.9$ in terms of the number of features and the classification performance, but AvePar included more different sizes of feature subsets. In five of the six datasets, BestPar achieved better classification performance and a smaller number of features than both $\gamma = 0.5$ and $\gamma = 0.9$, especially in the datasets with a larger number of features, such as the Statlog and Waveform datasets.

Figure 1 shows that MORSN can further reduce the number of features and increase the classification performance, which indicates that MORSN as a multi-objective approach can explore the search space of a feature selection problem better than the single objective algorithm, PSOPRSN.

MORSN Using NB and 5NN. The results of MORSN and PSOPRSN with $\gamma = 0.5$ and $\gamma = 0.9$ using 5NN and NB show similar patterns to those of using DT. In most cases, MORSN selected a smaller feature subset and decreased the classification error rate over using all features. MORSN outperformed PSOPRSN in terms of both the number of features and the classification performance, especially on the

datasets with a large number of features. The detailed descriptions and discussions are not presented to save space.

Note that, the results also show that the performance of MORSN and PSOPRSN are consistent when using different classification algorithms, which suggests that MORSN and PSOPRSN with probabilistic rough set as the evaluation criteria are general to these three classification algorithms.

5.3. *Results of MORSE*

Figure 2 shows the experimental results of MORSE and PSOPRSE on the test sets, where DT was used as the classification algorithm.

Results of MORSE Using DT. According to Fig. 2, in almost all cases (except for the Waveform dataset), the average Pareto front, MORSE-AvePar contains more than two solutions, which included a smaller size of feature subset and maintained or even increased the classification performance over using the full set of features. In *all* datasets, MORSE-BestPar obtained at least one feature subset, which included less than one third of the features and decreased the classification error rate of using all the available features. For example, in the Waveform dataset, MORSE-BestPar included a feature subset with only 8 features from the available 40 features. With the selected 8 features, DT obtained higher classification accuracy than with all the 40 features. The results suggest that MORSE as a multi-objective feature selection algorithm guided by the two objectives is able to explore the Pareto front effectively to select small feature subsets and obtain better classification performance than using all the available features.

Comparisons Between MORSE and PSOPRSE Using DT. According to Fig. 2, in *all* cases, MORSE-AvePar achieved similar or better results than PSOPRSE. MORSE-BestPar outperformed PSOPRSE in terms of both the number of features and the classification performance. In particular, in the Waveform dataset, the numbers of features in PSOPRSE are around 10 and around 27, which means in some runs, PSOPRSE is stagnation in local optima of having a large number of features (around 27). MORSE as a multi-objective algorithm, can overcome this problem, and all the feature subsets have less than 10 features. This suggests that MORSE as a multi-objective algorithm can better explore the solution space of a feature selection problem to achieve more and better solutions than the single objective algorithm, PSOPRSE.

MORSE and PSOPRSE Using NB and 5NN. In almost all cases, NB and 5NN using the feature subsets selected by MORSE achieved a similar or lower classification error rate than using the full set of features. MORSE outperformed PSOPRSE regarding the size of the feature subsets and the classification performance. This further shows the superior performance of the multi-objective algorithm, MORSE, over the single objective method, PSOPRSE. The results also suggest that MORSE and PSOPRSE show a similar pattern when using DT, NB or 5NN to evaluate the

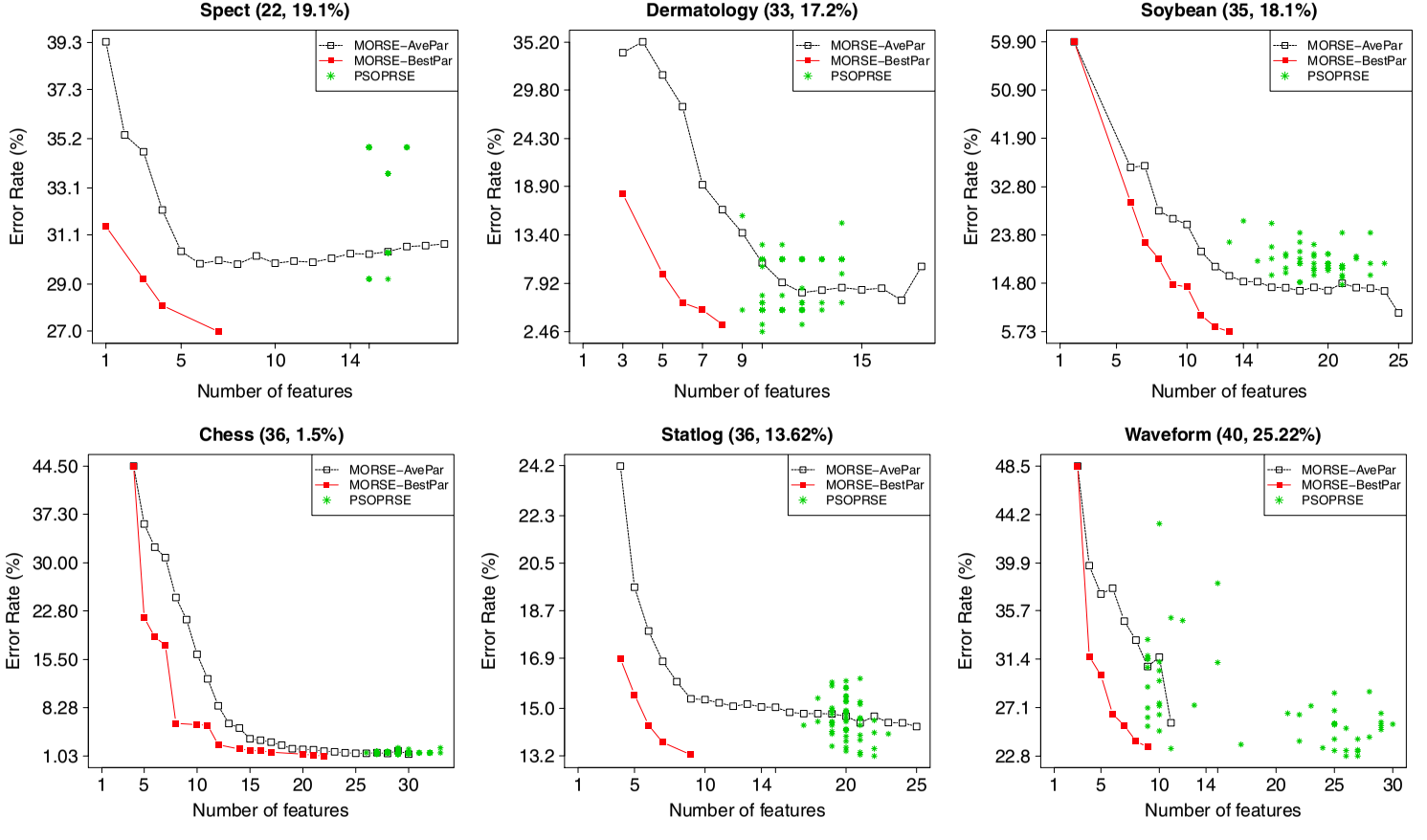

Fig. 2. Results of MORSE and PSOPRSE on test sets evaluated by DT.

classification error rate. This suggests that MORSE and PSOPRSE as filter feature selection algorithms are general to these three classification algorithms.

5.4. *Comparisons between MORSN and MORSE*

In this section, the results of MORSN and MORSE using DT as the classification algorithm are used as an example to compare the performance of MORSN and MORSE, which are shown in Fig. 3. The results of using NB and 5NN as the classification algorithms show similar patters as that of using DT.

According to Fig. 3, MORSN-AvePar and MORSE-AvePar achieved similar results in terms of the size and the classification performance in most cases, but MORSE-AvePar achieved a much lower classification error rate than MORSN-AvePar in the Dermatology and Soybean datasets. In most cases, MORSN-BestPar and MORSE-BestPar selected a similar number of features, but MORSE-BestPar obtained slightly better classification performance than MORSN-BestPar. In almost all cases, the lowest classification error rate is achieved by MORSE-BestPar.

MORSN and MORSE share the same parameter settings. The only difference is that MORSN uses the number of features as one of the two objectives while MORSE uses the number of equivalence classes to represent the number of features. Their different classification performance is mainly caused by the different evaluation criteria for the number of features. By further inspection and comparisons, we observe that the number of features selected by MORSN and MORSE are similar in most cases, but in almost all cases, they selected different combinations of individual features. Although MORSN selected a small number of features, these features can describe a large number of equivalence classes. There could be thousands of small equivalence classes, which only include one or two instances. If there is another equivalence class, which has slightly more instances, this class will dominate others and the obtained feature subsets will only contain information that can identify this particular class. Therefore, in this situation, without considering the size of the equivalence classes, the feature subsets selected by MORSN may lose generality and perform badly on unseen test data. Therefore, the classification performance of MORSE is usually better than MORSN.

5.5. *Comparisons with two traditional algorithms*

Table 3 shows the results of CfsF and CfsB for feature selection, where DT was used for classification. Comparing the results of the three single objective algorithms, PSOPRS, PSOPRSN and PSOPRSE with CfsF and CfsB, these three algorithms achieved better classification performance than CfsF and CfsB in five of the six datasets, although they selected a slightly larger number of features in some cases. In *all* datasets, the two multi-objective algorithms, MORSN and MORSE outperformed CfsF and CfsB in terms of the size of the feature subsets and the classification performance. The comparisons show that the five algorithms using PSO as the search

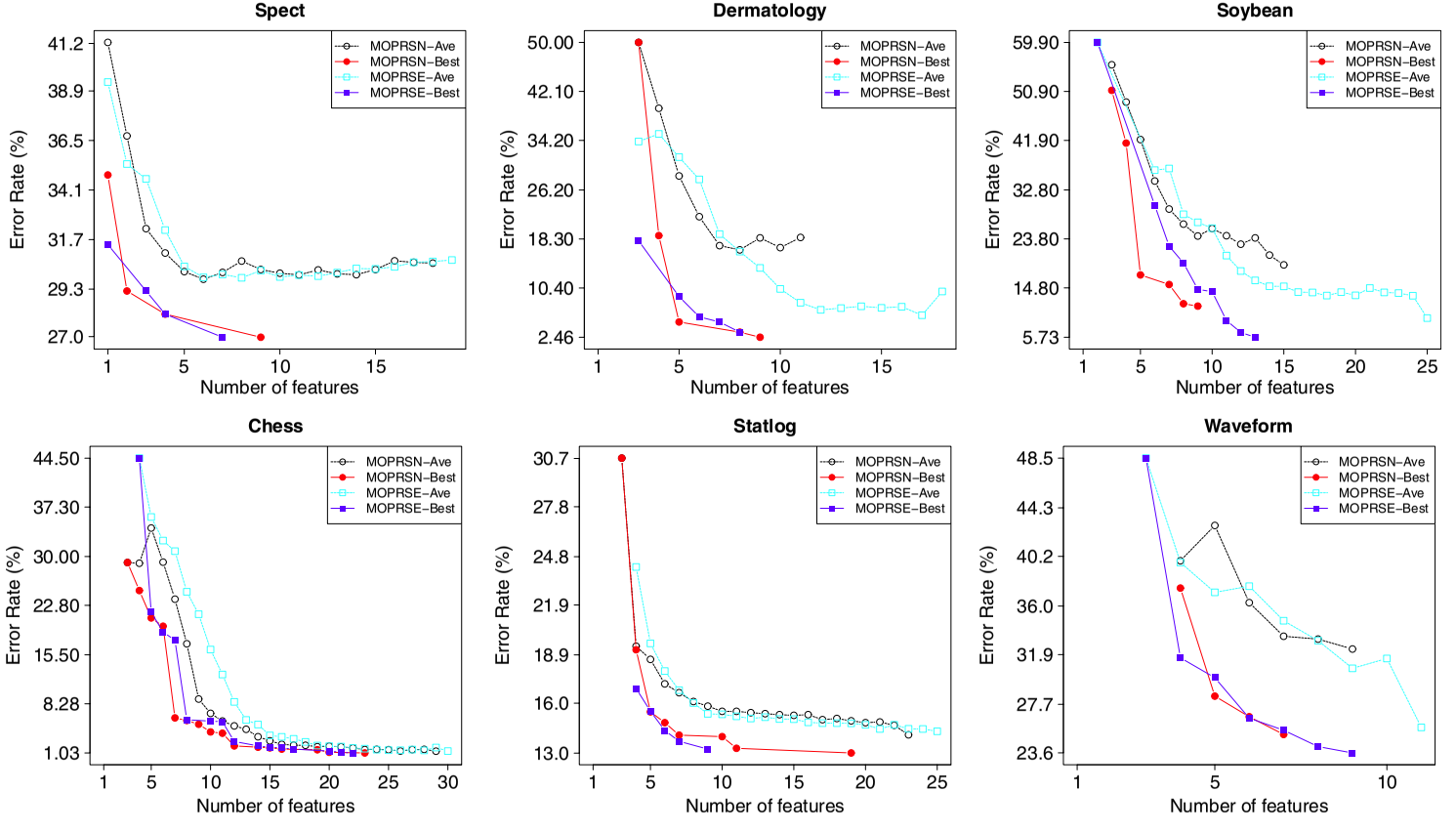

Fig. 3. Results of MORSN and MORSE on test sets evaluated by DT.

Table 3. Results of CfsF and CfsB with DT as the learning algorithm.

	Spect		Dermatology		Soybean		Chess		Statlog		Waveform	
Dataset Method	Size	Error (%)	Size	Error (%)	Size	Error (%)	Size	Error (%)	Size	Error (%)	Size	Error (%)
CfsF	4	30	17	12.73	12	19.51	5	21.9	5	28.38	32	28
CfsB	4	30	17	12.73	14	14.63	5	21.9	5	28.38	32	28

technique and using probabilistic rough set as the evaluation criteria can better solve the feature selection problems than CfsF and CfsB.

5.6. *AvePar versus BestPar*

Both AvePar and BestPar can show the performance of a multi-objective algorithm, but BestPar is a more appropriate way to present the results in feature selection tasks due to the following two reasons.

The first reason is that a solution in AvePar is not necessarily a complete/meaningful solution for a feature selection task. Each average solution is formed by the number m and the average classification error rate of all feature subsets of size m in the union set. However, feature selection problems do not only involve the number of features and the classification performance, but also involve the selected individual features. There can be many feature subsets with m features, but with different combinations of m features. So strictly speaking, the combinations of individual features cannot be averaged. Therefore, the solutions in AvePar is not a complete solution and should not be sent to users. The second reason is that BestPar involves a simple further selection process, which provides a better set of nondominated solutions to users. By selecting only the nondominated solutions from the union set, BestPar usually has a small number of solutions and the solutions usually have smaller numbers of features than AvePar solutions. It therefore provides fewer but better solutions to the users and reduces their cost for selecting a single solution. Meanwhile, each solution in BestPar is a complete solution of a feature selection problem. Multiple solutions with the same number of features and the same classification performance are presented at the same point in the figures, but all of them are complete solutions. Therefore, for a certain feature number m, BestPar could provide different combinations of individual features to users. Accordingly, BestPar is more appropriate than AvePar to show the performance of a multi-objective feature selection algorithm.

6. Further Experiments on Continuous Datasets

All the discrete datasets we can find in UCI and other rough set related papers[10,12–14] have a small number of features. To further test the performance of the five algorithms, we use a data discretization technique to pre-process the continuous data to

Table 4. Continuous datasets.

Dataset	# Features	# Classes	# Instances
Australian (Austral.)	14	2	690
German	24	2	1000
World Breast Cancer-Diagnostic (WBCD)	30	2	569
Ionosphere (Ionosph.)	34	2	351
Sonar	60	2	208
Musk Version 1 (Musk1)	166	2	476
Semeion	256	2	1593
Madelon	500	2	4400

discrete data. Any discretization technique can be used here. We choose a simple technique which is the filter discretization technique in Weka to make this process fast. The options in the filter discretization Weka is set as default. Eight continuous datasets listed in Table 4 were chosen from UCI and discretized. They were selected to have a large number of attributes (up to 500) and different numbers of classes and instances. Note that, after discretization, the classification performance of using all the discretized features on each dataset is still similar to that of using all the original continuous features. Since the results of using DT, NB and 5NN show similar patterns, only the results of DT are presented here. Table 5 shows the experimental results of the three single objective algorithms, PSOPRS, PSOPRSN and PSOPRSE. Figure 4 show the experimental results of MORSN and MORSE.

6.1. *Results of PSOPRS, PSOPRSN and PSOPRSE*

According to Table 5, it can be observed that in almost all cases, PSOPRS selected around two thirds of the available features and using the selected features, DT achieved similar or better (in most cases) classification performance than using all the original features. PSOPRSN further reduced the number of features and achieved similar (slightly better or worse) classification performance than using all the original features, which is worse than the classification performance of PSOPRS. In most cases, PSOPRSE maintain the classification performance achieved by PSOPRS, but further reduce the number of features selected. This is consistent with their results on the discrete datasets. The results suggest that the three single objective algorithms can also be successfully used for feature selection on the datasets with a large number of features.

6.2. *Results of MORSN and MORSE*

According to Fig. 4, we can observe that in most cases, the average Pareto fronts of MORSN (MORSN-Ave) and MORSE (MORSE-Ave) included a smaller number of features. DT using the small number of features improved the better classification performance over using all the available features. In *all* datasets, MORSN-Best and

Table 5. Results of PSOPRS, PSOPRSN ($\gamma = 0.9$ and $\gamma = 0.5$) and PSOPRSE.

Dataset	Method	Size	Best	Mean ± StdDev	Test	Dataset	Method	Size	Best	Mean ± StdDev	Test
Austral.	All	14	11.74		−	German	All	24	27.03		−
	PSOPRS	11.73	13.91	15.14 ± 1.08E0	+		PSOPRS	17.13	25.83	28.31 ± 1.38E0	=
	$\gamma = 0.9$	8	13.91	13.91 ± 30E-4	=		$\gamma = 0.9$	8.9	24.02	27.55 ± 1.79E0	=
	$\gamma = 0.5$	2	14.78	14.78 ± 26E-4	+		$\gamma = 0.5$	6.47	26.13	28.27 ± 1.11E0	=
	PSOPRSE	10	13.91	13.94 ± 15.6E-2			PSOPRSE	13.47	24.92	28.28 ± 1.63E0	
WBCD	All	30	7.41		=	Ionosph.	All	34	11.97		=
	PSOPRS	18.83	3.17	6.1 ± 1.4E0	−		PSOPRS	21.1	5.98	12.05 ± 3.33E0	=
	$\gamma = 0.9$	5.87	3.7	6.24 ± 1.65E0	=		$\gamma = 0.9$	5.03	6.84	15.95 ± 4.39E0	+
	$\gamma = 0.5$	4.13	3.7	5.54 ± 1.42E0	−		$\gamma = 0.5$	4.03	6.84	15.16 ± 4.04E0	=
	PSOPRSE	9.07	4.23	7.18 ± 1.59E0			PSOPRSE	6.63	7.69	13.11 ± 3.36E0	
Musk1	All	166	29.75		=	Sonar	All	60	31.88		+
	PSOPRS	101.1	22.15	27.78 ± 3.03E0	=		PSOPRS	36.13	18.84	25.7 ± 4.3E0	=
	$\gamma = 0.9$	44.77	22.78	28.86 ± 3.59E0	=		$\gamma = 0.9$	8.23	17.39	32.17 ± 6.52E0	+
	$\gamma = 0.5$	44.77	22.78	28.86 ± 3.59E0	=		$\gamma = 0.5$	8.17	18.84	32.56 ± 5.95E0	+
	PSOPRSE	81.13	23.42	29.66 ± 4.14E0			PSOPRSE	36.13	18.84	25.7 ± 4.3E0	
Semeion	All	256	5.65		−	Madelon	All	500	37.64		+
	PSOPRS	159.67	5.65	7.49 ± 85.8E-2	=		PSOPRS	301.97	17.09	24.48 ± 6.65E0	=
	$\gamma = 0.9$	84.07	5.08	7.65 ± 94.8E-2	=		$\gamma = 0.9$	183.43	17.32	33.27 ± 7.74E0	+
	$\gamma = 0.5$	84.07	5.08	7.65 ± 94.8E-2	=		$\gamma = 0.5$	183.43	17.32	33.27 ± 7.74E0	+
	PSOPRSE	143.07	5.65	7.73 ± 84.5E-2			PSOPRSE	301.97	17.09	24.48 ± 6.65E0	

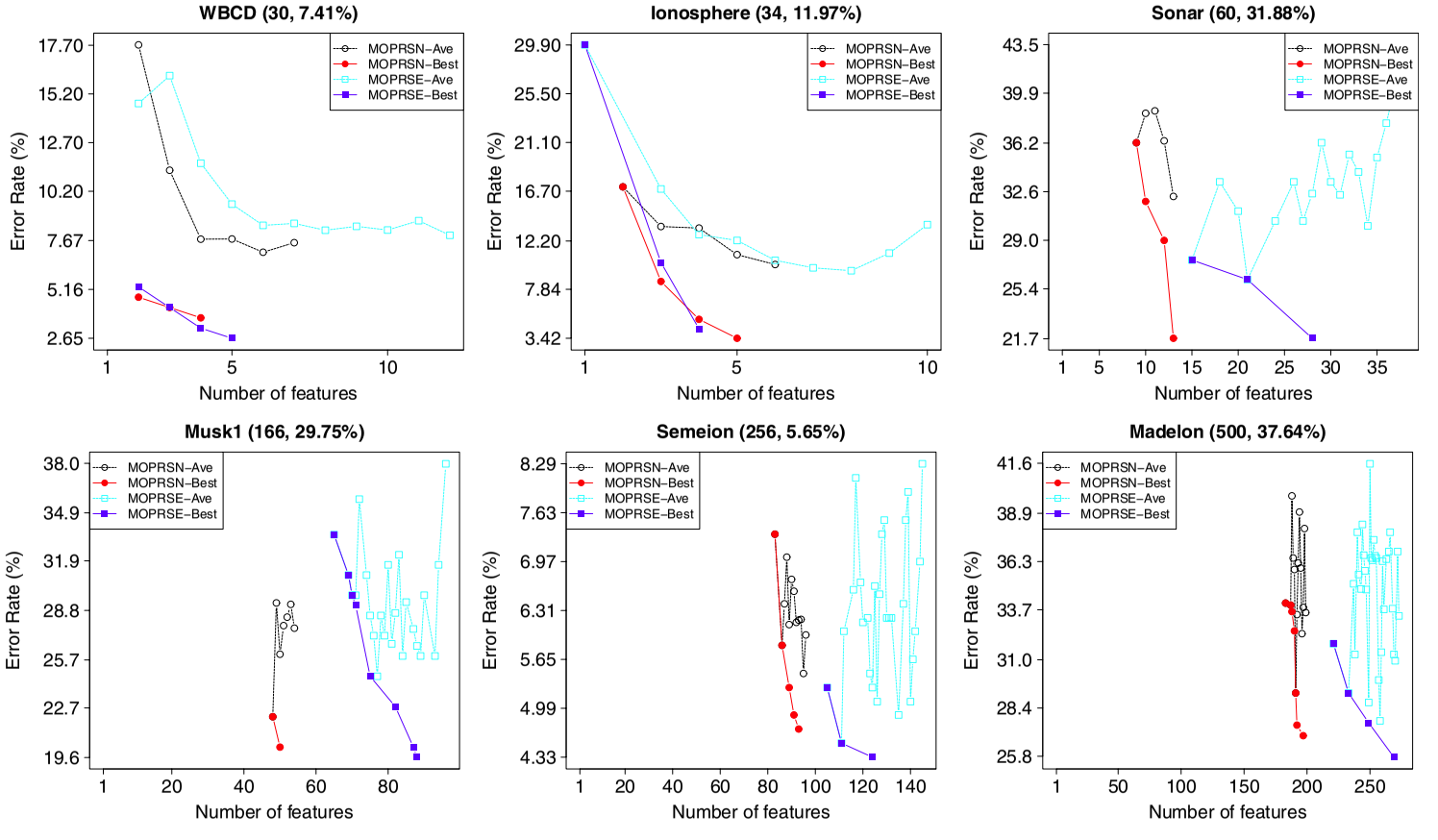

Fig. 4. Results of MORSN and MORSE on test sets evaluated by DT.

MORSE-Best achieved better classification performance than using all the original features. In most cases, MORSE-Ave achieved slightly better classification performance than MORSN-Ave and MORSE achieved better classification performance than MORSN, although the number of features in MORSE is slightly larger than MORSN. This is consistent with the results on the discrete datasets and our hypothesis in Sec. 3.2.

Comparing the results in Fig. 4 with that in Table 5, it can be seen that in almost all cases, MORSN and MORSE outperformed PSOPRS, PSOPRSN and PSOPRSE in terms of both the size of the selected feature subsets and the classification performance. The results suggest that both MORSN and MORSE can be successfully applied to address feature selection problems on the discretized continuous datasets with a large number of features.

The results also show that the performance of PSOPRS, PSOPRSN PSOPRSE, MORSN and MORSE are general to the three different classification algorithms (DT, NB and 5NN). This further demonstrated that these five filter algorithms are general to the three different classification algorithms.

Note that, the classification performance presented in Table 5 and Fig. 4 were obtained by using the selected features on the discretized continuous datasets. We also further tested the classification performance of the selected features on the original continuous datasets and the results show that in most cases, the three classification algorithms using the selected features (in continuous data) can achieve similar or even better classification performance than using all the continuous features. This indicates that although PSOPRS, PSOPRSN PSOPRSE, MORSN and MORSE were designed for discrete datasets, they can be easily used for continuous datasets by a simple discretization step.

6.3. *Further comparisons with existing methods*

To further investigate the performance of the proposed algorithms, three existing feature selection algorithms, including two single objective filter algorithms,[34,35] and a filter-based multi-objective algorithm (CMDfsE),[41] are used for comparisons.

The two single objective algorithms used fuzzy set theory with PSO[35] and with GA[34] for feature selection, where one of the two datasets used in the experiments is the Sonar dataset. Comparing the results on the Sonar dataset, MORSE achieved better classification performance than the two algorithms proposed in the literatures.[34,35]

CMDfsE[41] is a filter-based multi-objective algorithm using PSO and information theory. There are four datasets (Spect, Dermatology, Soybean and Chess) used in both this paper and in the literature.[41] Comparing the results, it can be observed that MORSE generally achieved similar performance to that of CMDfsE in terms of both the classification performance and the number of features, but the graphs presenting the results of AvePar and BestPar in MORSE are less varied than that of CMDfsE.

7. Conclusion

The overall goal of this paper was to propose a filter-based multi-objective feature selection approach based on PSO and PRS. The goal was successfully achieved by developing two filter-based multi-objective methods (MORSN and MORSE). PSO as a powerful global search technique is considered to address the main challenge of having a large search space in feature selection problems. More importantly, the employed multi-objective PSO algorithm in MORSN and MORSE uses mutation operators and a crowding distance measure, which can maintain the diversity of the swarm to avoid premature convergence. This is highly important in feature selection problems, where the fitness landscape has many local optima. Meanwhile, PRS can properly measure the relevance between a group of features and the class labels, which is a key factor in filter feature selection approaches. The powerful search ability of the multi-objective PSO and the proper PRS-based measure lead to the good performance of MORSN and MORSE, which outperformed a new single objective algorithm, two existing single objective algorithms and two traditional methods. Furthermore, the new PRS-based measure for minimization of the number of features in MORSE considers the number of equivalence classes, which can avoid the problem of selecting a small feature subset but losing generality. This measure leads to the better classification performance in MORSE than in MORSN. The results on both discrete datasets and the continuous datasets with a large number of features demonstrate that the proposed algorithms as filter approaches are general to the different classification algorithms (i.e., DT, NB and 5NN).

This study demonstrates that multi-objective PSO and PRS can address feature selection problems to obtain a set of nondominated solutions more effectively than a single solution generated by the three single objective algorithms. This work also highlights that when using PRS for feature selection, considering the number of equivalence classes instead of the number of features, can further increase the classification performance without significantly increasing the size of the selected feature subset. Moreover, the use of continuous datasets in the experiments not only shows that the proposed algorithms can be applied to problems with a large number of features, but also suggests that rough set theory can function well on such large scale problems. The observations from this research show the success of using PSO and PRS on feature selection problems. In future, we will further explore the potential of PSO and PRS to better address feature selection tasks.

Acknowledgments

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), the Marsden Funds of New Zealand (VUW1209 and VUW0806) and the University Research Funds of Victoria University of Wellington (203936/3337, 200457/3230).

References

1. I. Guyon and A. Elisseeff, An introduction to variable and feature selection, *J. Mach. Learn. Res.* **3** (2003) 1157–1182.
2. M. Dash and H. Liu, Feature selection for classification, *Intell. Data Anal.* **1**(4) (1997) 131–156.
3. J. Kennedy and R. Eberhart, Particle swarm optimization, *IEEE Int. Conf. Neural Networks*, The University of Western Australia, Perth, Western Australia, Vol. 4, pp. 1942–1948, 1995.
4. Y. Shi and R. Eberhart, A modified particle swarm optimizer, *IEEE Int. Conf. Evolutionary Computation (CEC'98)*, Anchorage, Alaska, USA, pp. 69–73, 1998.
5. J. Kennedy and W. Spears, Matching algorithms to problems: An experimental test of the particle swarm and some genetic algorithms on the multimodal problem generator, in *IEEE Congr. Evolutionary Computation (CEC'98)*, Anchorage, Alaska, USA, pp. 78–83, 1998.
6. Y. Liu, G. Wang, H. Chen and H. Dong, An improved particle swarm optimization for feature selection, *J. Bionic Eng.* **8**(2) (2011) 191–200.
7. L. Cervante, B. Xue, M. Zhang and L. Shang, Binary particle swarm optimisation for feature selection: A filter based approach, in *IEEE Congr. Evolutionary Computation (CEC'2012)*, pp. 881–888, 2012.
8. I. A. Gheyas and L. S. Smith, Feature subset selection in large dimensionality domains, *Pattern Recogn.* **43**(1) (2010) 5–13.
9. Z. Pawlak, Rough sets, *Int. J. Parallel Program.* **11** (1982) 341–356.
10. X. Wang, J. Yang, X. Teng, W. Xia and R. Jensen, Feature selection based on rough sets and particle swarm optimization, *Pattern Recogn. Lett.* **28**(4) (2007) 459–471.
11. Y. Yao and Y. Zhao, Attribute reduction in decision-theoretic rough set models, *Inf. Sci.* **178**(17) (2008) 3356–3373.
12. C. Bae, W.-C. Yeh, Y. Y. Chung and S.-L. Liu, Feature selection with intelligent dynamic swarm and rough set, *Expert Syst. Appl.* **37**(10) (2010) 7026–7032.
13. Y. Chen, D. Miao and R. Wang, A rough set approach to feature selection based on ant colony optimization, *Pattern Recogn. Lett.* **31**(3) (2010) 226–233.
14. L. Cervante, B. Xue, L. Shang and M. Zhang, A dimension reduction approach to classification based on particle swarm optimisation and rough set theory, in *25th Australasian Joint Conf. Artificial Intelligence,* Lecture Notes in Computer Science, Vol. 7691 (Springer, 2012), pp. 313–325.
15. L. Cervante, B. Xue, L. Shang and M. Zhang, A multi-objective feature selection approach based on binary pso and rough set theory, in *13th European Conf. Evolutionary Computation in Combinatorial Optimization (EvoCOP)*, Lecture Notes in Computer Science, Vol. 7832 (Springer, 2013), pp. 25–36.
16. L. Cervante, B. Xue, L. Shang and M. Zhang, Binary particle swarm optimisation and rough set theory for dimension reduction in classification, in *IEEE Congr. Evolutionary Computation (CEC'13)*, Cancun, Mexico, pp. 2428–2435, 2013.
17. J. Kennedy and R. Eberhart, A discrete binary version of the particle swarm algorithm, in *IEEE Int. Conf. Systems, Man, and Cybernetics, 1997, Computational Cybernetics and Simulation*, Orlando, Florida, USA, Vol. 5, pp. 4104–4108, 1997.
18. A. Whitney, A direct method of nonparametric measurement selection, *IEEE Trans. Comput.* **C-20**(9) (1971) 1100–1103.
19. T. Marill and D. Green, On the effectiveness of receptors in recognition systems, *IEEE Trans. Inf. Theory* **9**(1) (1963) 11–17.
20. S. Stearns, On selecting features for pattern classifier, in *Proc. 3rd Int. Conf. Pattern Recognition* (Coronado, Calif, USA), pp. 71–75, 1976.

21. P. Pudil, J. Novovicova and J. V. Kittler, Floating search methods in feature selection, *Pattern Recogn. Lett.* **15**(11) (1994) 1119–1125.
22. L. Oliveira, R. Sabourin, F. Bortolozzi and C. Suen, Feature selection using multi-objective genetic algorithms for handwritten digit recognition, in *16th Int. Conf. Pattern Recognition (ICPR'02)*, Quebec City, Canada, Vol. 1, pp. 568–571, 2002.
23. Z. X. Zhu, Y. S. Ong and M. Dash, Wrapper-filter feature selection algorithm using a memetic framework, *IEEE Trans. Syst. Man Cybern. B, Cybern.* **37**(1) (2007) 70–76.
24. K. Neshatian and M. Zhang, Dimensionality reduction in face detection: A genetic programming approach, in *24th Int. Conf. Image and Vision Computing New Zealand (IVCNZ'09)*, pp. 391–396, 2009.
25. K. Neshatian, M. Zhang and P. Andreae, Genetic programming for feature ranking in classification problems, in *Simulated Evolution and Learning*, Lecture Notes in Computer Science, Vol. 5361 (Springer, Berlin/Heidelberg, 2008), pp. 544–554.
26. H. R. Kanan and K. Faez, An improved feature selection method based on ant colony optimization (aco) evaluated on face recognition system, *Appl. Math. Comput.* **205**(2) (2008) 716–725.
27. Y. Marinakis, M. Marinaki and G. Dounias, Particle swarm optimization for pap-smear diagnosis, *Expert Syst. Appl.* **35**(4) (2008) 1645–1656.
28. C. L. Huang and J. F. Dun, A distributed PSO-SVM hybrid system with feature selection and parameter optimization, *Appl. Soft Comput.* **8** (2008) 1381–1391.
29. R. Fdhila, T. Hamdani and A. Alimi, Distributed mopso with a new population subdivision technique for the feature selection, in *5th Int. Symp. Computational Intelligence and Intelligent Informatics (ISCIII 2011)*, Florida, USA, pp. 81–86, 2011.
30. L. Y. Chuang, H. W. Chang, C. J. Tu and C. H. Yang, Improved binary PSO for feature selection using gene expression data, *Comput. Biol. Chem.* **32**(29) (2008) 29–38.
31. M. A. Hall, Correlation-based feature subset selection for machine learning, PhD thesis, The University of Waikato, Hamilton, New Zealand, 1999.
32. H. Almuallim and T. G. Dietterich, Learning boolean concepts in the presence of many irrelevant features, *Artif. Intell.* **69** (1994) 279–305.
33. K. Kira and L. A. Rendell, A practical approach to feature selection, *Assorted Conf. and Workshops*, Aberdeen, Scotland, pp. 249–256, 1992.
34. B. Chakraborty, Genetic algorithm with fuzzy fitness function for feature selection, in *IEEE Int. Symp. Industrial Electronics (ISIE'02)*, L'Aquila, Italy, Vol. 1, pp. 315–319, 2002.
35. B. Chakraborty, Feature subset selection by particle swarm optimization with fuzzy fitness function, in *3rd Int. Conf. Intelligent System and Knowledge Engineering (ISKE'08)*, Xiamen, China, Vol. 1, pp. 1038–1042, 2008.
36. K. Neshatian and M. Zhang, Pareto front feature selection: Using genetic programming to explore feature space, in *Proc. 11th Annual Conf. Genetic and Evolutionary Computation (GECCO'09)*, Montreal, Canada, pp. 1027–1034, 2009.
37. K. Iswandy and A. Koenig, Feature-level fusion by multi-objective binary particle swarm based unbiased feature selection for optimized sensor system design, in *IEEE Int. Conf. Multisensor Fusion and Integration for Intelligent Systems*, Heidelberg, Germany, pp. 365–370, 2006.
38. M. R. Sierra and C. A. C. Coello, Improving PSO-based multi-objective optimization using crowding, mutation and epsilon-dominance, in *Proc. Third Int. Conf. Evolutionary Multi-Criterion Optimization*, Guanajuato, Mexico, pp. 505–519, 2005.
39. K. Bache and M. Lichman, UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science (2013).

40. I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd edn. (Morgan Kaufmann, 2005).
41. B. Xue, L. Cervante, L. Shang, W. N. Browne and M. Zhang, A multi-objective particle swarm optimisation for filter based feature selection in classification problems, *Connect. Sci.* **24** (2012) 91–116.