A Multi-objective Feature Selection Approach Based on Binary PSO and Rough Set Theory

Liam Cervante[1], Bing Xue[1], Lin Shang[2], and Mengjie Zhang[1]

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

Abstract. Feature selection has two main objectives of maximising the classification performance and minimising the number of features. However, most existing feature selection algorithms are single objective wrapper approaches. In this work, we propose a multi-objective filter feature selection algorithm based on binary particle swarm optimisation (PSO) and probabilistic rough set theory. The proposed algorithm is compared with other five feature selection methods, including three PSO based single objective methods and two traditional methods. Three classification algorithms (naïve bayes, decision trees and k-nearest neighbours) are used to test the generality of the proposed filter algorithm. Experiments have been conducted on six datasets of varying difficulty. Experimental results show that the proposed algorithm can automatically evolve a set of non-dominated feature subsets. In almost all cases, the proposed algorithm outperforms the other five algorithms in terms of both the number of features and the classification performance (evaluated by all the three classification algorithms). This paper presents the first study on using PSO and rough set theory for multi-objective feature selection.

Keywords: Particle Swarm Optimisation, Feature Selection, Rough Set Theory, Multi-objective Optimisation.

1 Introduction

Classification tasks are to classify a given instance in the dataset to one of the known classes according to the information described by features. However, some of them are irrelevant or redundant features, which may even increase the classification error rate. Feature selection is to select a subset of relevant features to achieve similar or even better classification performance [6]. By reducing or eliminating the irrelevant and redundant features, feature selection can reduce the dimensionality of the data, simplify the learnt classifier, reduce the training time, and/or increase the classification accuracy [4,17].

Based on the evaluation criteria, feature selection methods are generally classified into two categories: wrapper and filter approaches [6]. Wrapper approaches

M. Middendorf and C. Blum (Eds.): EvoCOP 2013, LNCS 7832, pp. 25–36, 2013.

include a learning/classification algorithm in the evaluation procedure while filter approaches do not. Therefore, wrappers usually achieve better results than filter approaches, but they are computationally expensive. Filter approaches are more general and computationally cheaper than wrapper approaches, but an appropriate evaluation criterion is needed in filter approaches [4,6].

A challenge in feature selection is that the size of the search space is 2^n, where n is the total number of features. Most of the existing feature selection algorithms suffer from the problems of stagnation in local optima and high computational cost [4,17], especially for wrapper approaches. Evolutionary computation (EC) techniques are well-known for their global search ability. Particle swarm optimisation (PSO) [14] is a relatively recent EC technique, which is computationally less expensive than other EC algorithms. Therefore, PSO has recently gained more attention for solving feature selection problems [17,11].

Feature selection is a multi-objective problem, which aims to maximise the classification performance and minimise the number of features selected. However, most of the existing EC based feature selection algorithms are wrapper based single objective approaches. The use of wrapper algorithms is limited in real-world applications because of their high computational cost. Meanwhile, from a theoretical point of view, Yao and Zhao [21] have shown that probabilistic rough set can be a good measure in feature selection. Therefore, it is thought to develop a filter based multi-objective feature selection approach using PSO and probabilistic rough set theory.

1.1 Goals

The overall goal of this paper is to develop a filter based multi-objective feature selection approach to obtaining a set of non-dominated solutions, which include a smaller number of features and achieve similar or even better classification performance than using all features. To achieve this goal, we propose a multi-objective feature selection algorithm based on PSO and probabilistic rough set theory. Specifically, we will investigate

- whether using PSO and *probabilistic* rough set theory can reduce the number of features and maintain or even increase the classification performance, and can outperform the algorithm using PSO and *standard* rough set theory,
- whether considering *the number of features* in the fitness function can further reduce the number of features and maintain the classification performance,
- whether the proposed *multi-objective* algorithm can obtain a set of non-dominated feature subsets, and can outperform two traditional methods and the above three single objective methods, and
- whether the proposed algorithm is general to different learning algorithms.

2 Background

2.1 Binary Particle Swarm Optimisation

In PSO [14], a particle represents a candidate solution. Particles move in the D-dimensional search space to search for the best solutions. Particle i has a

position denoted by $x_i = (x_{i1}, x_{i2}, ..., x_{iD})$ and a velocity denoted by $v_i = (v_{i1}, v_{i2}, ..., v_{iD})$. During the search process, the best position visited so far by the particle is its personal best (*pbest*) and the best position obtained by the population thus far is called global best (*gbest*). Particles share information through *pbest* and *gbest* to update their positions and velocities to search for the optimal solutions. In binary PSO (BPSO) [9], x_i, *pbest* and *gbest* are restricted to 1 or 0. The position and velocity updating equations can be seen as follows:

$$v_{id}^{t+1} = w * v_{id}^{t} + c_1 * r_1 * (p_{id} - x_{id}^{t}) + c_2 * r_2 * (p_{gd} - x_{id}^{t}) \tag{1}$$

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

where t represents the tth iteration in the evolutionary process. $d \in D$ represents the dth dimension in the search space. w is the inertia weight. c_1 and c_2 are acceleration constants. r_1 and r_2 are random constants uniformly distributed in [0, 1]. p_{id} and p_{gd} denote the values of *pbest* and *gbest* in the dth dimension. $rand()$ is a random number selected from a uniform distribution in [0,1].

2.2 Rough Set Theory

Rough set theory [13] is an intelligent mathematical tool to handle uncertainty, imprecision and vagueness. One of the strengths of rough set theory is that it does not need any prior knowledge about data.

In rough set theory, knowledge and information is represented as an information system, which can be denoted as $I = (U, A)$, where U is a finite non-empty set of objects and A is the attributes/features that describe each object. For any $P \subseteq A$ and $X \subseteq U$, there is an equivalence relation defined as $IND(P) = \{(x, y) \in U * U | \forall a \in P, a(x) = a(y)\}$. If two objects in U satisfy $IND(P)$, they are indiscernible with regards to P. The equivalence relation $IND(P)$ induces a partition of U denoted by U/P, which induces the equivalence classes. The equivalence class of U/P containing x is given by $[x]_P = [x]_A = y \in U | (x, y) \in IND(P)$. The equivalence classes are the basic blocks to construct rough set approximations. For $X \subset U$, a lower approximation $\underline{P}X$ and an upper approximation $\overline{P}X$ of X with respect to $IND(P)$ are defined as follows [13]:

$$\underline{P}X = \{x \in U | [x]_P \subseteq X\} \qquad \overline{P}X = \{x \in U | [x]_P \cap X \neq \emptyset\} \tag{3}$$

$\underline{P}X$ contains all the objects, which are surely belong to the target set X. $\overline{P}X$ contains the objects, which are surely or probably belong to the target set X.

An ordered pair $(\overline{P}X, \underline{P}X)$ is called a rough set. The concept of the reduct is fundamental in rough sets theory. A reduct is the essential part of $I = (U, A)$, which can achieve similar approximation power of classification as all the original features A. There could be many different reducts and feature selection using rough set theory is to remove redundant and irrelevant features to search for the smallest reduct (feature subset).

$\underline{P}X$ and $\overline{P}X$ in standard rough set theory were defined as two extreme cases in terms of the relationship of a equivalence class and the target set [13]. The degree of their overlap is not taken into account, which will unnecessarily limit its applications. Therefore, researchers investigate probabilistic rough set theory to relax the definitions of the lower and upper approximation [21]. The lower approximation is re-defined as Equation 4, where $\mu_P[x]$ shown is defined as a way to measure the fitness of a given instance $x \in X$.

$$\underline{apr}_P X = \{x | \mu_P[x] \geq \alpha\} \tag{4}$$

where

$$\mu_P[x] = \frac{|[x]_P \cap X|}{|[x]_P|} \tag{5}$$

α can be adjusted to restrict or relax the lower approximation. If a large number of instances X are in the target set but a small number are not in a given equivalence class, it can include them in the lower approximation. $\underline{apr}_P X = \underline{P}X$ when $\alpha = 1$.

From theoretical point of view, Yao and Zhao have claimed that probabilistic rough set can be a good way for feature selection problems [21]. However, it has not been proved by any experiment.

2.3 Related Work on Feature Selection

Traditional Feature Selection Approaches. Hall [7] proposes a filter feature selection method (Cfs) based on the correlation between features and class labels. FOCUS algorithm [1], 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 methods are greedy search based sequential forward selection (SFS) [19] and sequential backward selection (SBS) [10]. In SFS (SBS), once a feature is selected (eliminated) it cannot be eliminated (selected) later, which causes the problem of so-called nesting effect. The "plus-l-take away-r" method proposed by Stearns [16] could overcome this limitation by performing l times forward selection followed by r times backward elimination. However, the determination of the optimal values of (l, r) is a difficult problem.

EC Algorithms for Features Selection. EC techniques have been applied to address feature selection problems. Based on genetic algorithms (GAs), Chakraborty [3] proposes a feature selection algorithm using a fuzzy sets based fitness function. Kourosh and Zhang [12] propose a genetic programming based filter method as a multi-objective approach for feature selection in binary classification problems. Based on ant colony optimisation and fuzzy-rough theory, Jensen [8] proposes a filter feature selection method for web content classification and complex systems monitoring.

Unler and Murat [17] propose a PSO based feature selection algorithm with an adaptive selection strategy. Mohemmed et al. [11] propose a hybrid method that

incorporates PSO with an AdaBoost framework to search for the best feature subset and determine the decision thresholds of AdaBoost simultaneously. Wang et al. [18] propose a filter feature selection algorithm based on an improved binary PSO and rough set. However, the feature subset is only tested on one learning algorithm, which can not show the advantage that filter algorithms are more general.

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 probabilistic rough set theory for feature selection has not been investigated in multi-objective feature selection. Therefore, the development of using PSO and probabilistic rough set for multi-objective feature selection is still an open issue.

3 Proposed Multi-objective Method

In this section, three feature selection algorithms [2] based on PSO and probabilistic rough set theory is firstly described, which are used as the baseline to test the performance of the proposed algorithm. Then we propose a multi-objective algorithm (MOPSOPRS) based on PSO and probabilistic rough set theory.

3.1 PSORS, PSOPRS and PSOPRSN

When using rough set theory 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 rough set theory. Based on the equivalence described by A, U can be partitioned to $U_1, U_2, U_3, ..., U_n$, where n is the number of classes in the dataset. After feature selection, the achieved feature subset can be considered as $P \subseteq A$. 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.

PSORS. In standard rough set theory, for $U_1 \subseteq U$ and $P \subseteq A$, $\underline{P}U_1 = \{x \in U | [x]_P \subseteq U_1\}$ is the lower approximation of P according to U_1 if $[x]_P$ only contains instances in U_1. $\underline{P}U_1$ measures the number of instances that have been completely separated from instances of other classes. Therefore, how well P describes each target in U can be calculated by Equation 6, which is the fitness function in PSOPRS. A feature subset with $Fitness_1(P) = 1.0$ means this feature subset can completely separate each class from the other classes.

$$Fitness_1(P) = \frac{\sum_{i=1}^{n} |\underline{P}U_i|}{|\mathbb{U}|} \tag{6}$$

PSOPRS. As discussed in Section 2.2, the definitions of lower approximation and upper approximation limit the application of standard rough set theory. Therefore, a filter feature selection algorithm (PSOPRS) based on PSO and probabilistic rough set theory was proposed in [2]. In probabilistic rough set

Algorithm 1. Pseudo-Code of MOPSOPRS

```
begin
    initialise the set of leaders LeaderSet and Archive
    calculate the crowding distance of each member in LeaderSet;
    while Maximum Iteration is not reached do
        for each particle do
            select a leader (gbest) from LeaderSet for each particle by using a
            binary tournament selection based on the crowding distance;
            update the velocity and the position of particle i;
            apply bit-flip mutation;
            evaluate two objective values of each particle;        /* number of
            features and Fitness2(P) value of the feature subset) */
            update the pbest of each particle;
        identify the non-dominated solutions (particles) to update LeaderSet;
        send leaders to Archive;
        calculate the crowding distance of each member in LeaderSet;
    calculate the classification error rate of solutions in Archive on the test set;
    return the solutions in Archive and their training and test classification
    error rates;
```

theory, for the target set U_1, $\mu_P[x] = \frac{|[x]_P \cap U_1|}{|[x]_P|}$. $\mu_P[x]$ 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 rather than $\underline{P}U_1$. $[x]_P$ does not have to completely contained in U_1. α can be adjusted to restrict or relax $\underline{apr}_P U_1$. When $\alpha = 1.0$, $\underline{apr}_P U_1 = \underline{P}U_1$. The fitness function of PSOPRS is shown by Equation 7.

$$Fitness_2(P) = \frac{\sum_{i=1}^{n} |\underline{apr}_P U_i|}{|\mathbb{U}|} \tag{7}$$

PSOPRSN. PSOPRS using probabilistic rough set theory can avoid the problems caused by standard rough set, but the number of features is not considered in the fitness function. For the same α value, if there are more than one feature subsets that have the same fitness, PSOPRS does not intend to search for the smaller feature subset. Therefore, the number of features was added to the fitness function to form another algorithm (PSOPRSN) [2], which aims to maximise the representation power of the feature subset and also to minimise the number of features.

$$Fitness_3(P) = \gamma * \frac{\sum_{i=1}^{n} |\underline{apr}_P U_i|}{|\mathbb{U}|} + (1 - \gamma) * (1 - \frac{\#features}{\#totalFeatures}) \tag{8}$$

where $\gamma \in (0, 1]$ shows the relative importance of the representation power while $(1 - \gamma)$ shows the relative importance of the number of features.

Table 1. Datasets

Dataset	#Features	#Classes	#Instances	Dataset	#Features	#Classes	#Instances
Spect	22	2	267	Dermatology	33	6	366
Soybean Large	35	19	307	Chess	36	2	3196
Statlog	36	6	6435	Waveform	40	3	5000

3.2 MOPSOPRS

PSOPRSN combines the two main objectives of feature selection into a single fitness function. However, γ needs to be predefined and its best value is problem-dependent. Therefore, we propose a multi-objective PSO based feature selection algorithm. However, PSO was originally proposed for single objective optimisation. Sierra and Coello [15] proposed a multi-objective PSO based on the ideas of mutation, crowding and dominance, which is a continuous algorithm and has achieved good performance. In this work, we extend it to a binary version of multi-objective PSO based on which we propose a multi-objective feature selection algorithm using probabilistic rough set theory (MOPSOPRS). The two objectives in MOPSOPRS is to maximise the representation power of the feature subset evaluated by $Fitness_2$ and to minimise the number of features.

Algorithm 1 shows the pseudo-code of MOPSOPRS. To select a *gbest* for each particle, MOPSOPRS employs a leader set to store the non-dominated solutions as the potential leaders. A crowding factor is employed to decide which non-dominated solutions should be added into the leader set and kept during the evolutionary process. A binary tournament selection is used to select two solutions from the leader set and the less crowded solution is chosen as the *gbest*. The maximum number of non-dominated solutions in the leader set is usually set as the same as the population size. In order to keep the diversity of the swarm and improve the search ability of the algorithm, MOPSOPRS randomly divides the whole swarm into three different groups in the initialisation procedure. The first group does not have any mutation. The second group employs uniform mutation to keep the global search ability and the third group employs non-uniform mutation to keep the local search ability. Furthermore, the three groups have the same leader set, which allows them to share their success to take advantages of different behaviors to search for the Pareto non-dominated solutions.

In all the algorithms, the dimensionality of the search space is the total number of features. Each particle is encoded in a binary string, where the "1" means the corresponding feature is selected, otherwise the feature is removed.

4 Experimental Design

As rough set theory only works on discrete values, six categorical datasets (Table 1) of varying difficulty are chosen from UCI machine learning repository [5] to test the performance of the algorithms. In each dataset, 70% of the instances are randomly chosen as the training set and others (30%) are the test set.

Table 2. Results of PSOPRS and PSOPRS with DT as the learning algorithm

Dataset	Spect			Dermatology			Chess		
Method	Size	Ave(Min)	Std	Size	Ave(Min)	Std	Size	Ave(Min)	Std
All	22	19.1		33	17.21		36	1.5	
PSORS	17.5	19.03(15.73)	2.28	21	13.99(2.46)	4.76	30.8	1.68(1.31)	0.261
PSOPRS									
$\alpha = 0.9$	17.3	19.44(15.73)	2.21	21	13.99(2.46)	4.76	30.7	1.6(1.31)	0.221
$\alpha = 0.8$	17.5	19.96(17.98)	1.96	21	13.99(2.46)	4.76	29.97	1.72(1.5)	0.279
$\alpha = 0.75$	15.57	18.2(17.98)	0.841	21	13.99(2.46)	4.76	30.3	1.53(1.31)	0.129
$\alpha = 0.5$	16.6	19.96(15.73)	2.11	20.73	13.99(2.46)	5.07	28.8	1.9(1.31)	0.525

All the algorithms firstly run on the training set to select a feature subset(s). The classification performance of the selected feature subset(s) will be evaluated by a learning/classification algorithm on the unseen test set. To test the claim that filter feature selection methods are general, three different learning algorithms, DT, naïve bayes (NB) and KNN with K=5 (5NN), are used in the experiments.

In all algorithms, the fully connected topology is used, the maximum velocity $v_{max} = 6.0$, the population size is 30 and the maximum iteration is 500. $w = 0.7298, c_1 = c_2 = 1.49618$. These values are chosen based on the common settings in the literature [14]. The algorithm has been conducted for 30 independent runs on each dataset. In PSOPRS, five different α values (1.0, 0.9, 0.8, 0.75, 0.5) are used in the experiments. When $\alpha = 1$, PSOPRS is the same as PSORS. Therefore, the results of $\alpha = 1$ in PSOPRS is not presented in Section 5. In PSOPRSN, the results of $\gamma = 0.9$ and $\gamma = 0.5$ are used to compare with that of the multi-objective algorithm (MOPSOPRS).

To further examine the performance of MOPSOPRS, two conventional filter feature selection methods (CfsF and CfsB) [7] implemented in Weka [20] are used for comparison and the classification performance is calculated by DT.

5 Experimental Results and Discussions

5.1 Experimental Results of PSORS and PSOPRS

Experiments about PSORS and PSOPRS have been conducted on the six datasets and DT, NB and 5NN were used for classification on the test sets. Due to the page limit, only the results of three datasets (Spect, Dermatology and Chess) using DT for classification are presented in Table 2. In the table, "All" means that all of the available features are used for classification. "Size" means the average number of features selected in the 30 independent runs. "Ave", "Min" and "Std" represent the average, the lowest and the standard deviation of the classification error rates achieved by DT across the 30 runs.

Results of PSORS. According to Table 2, in most cases, PSORS selected feature subsets, which included around two thirds of the available features and achieved similar classification performance to all features. In almost all datasets, the best classification performance of PSOPRS (Min) is better than all features. The results suggest that PSORS based on PSO and standard rough set theory can be successfully used for feature selection.

Results of PSOPRS. According to Table 2, in most cases, PSOPRS with different α can achieve similar classification performance to all features. The number of features generally decreases when α reduces. Meanwhile, the best results achieved by PSOPRS are always better than all features in all cases. The results suggests that by using probabilistic rough set for feature selection, PSOPRS can further reducing the number of features without reducing its classification performance. A smaller α means more relax on the lower and upper approximations, which usually can slightly remove more unnecessary features to further reduce dimensionality of the datasets.

Note that considering *all* experimental results on PSOPRS (not only the results in Table 2), in most cases, $\alpha = 0.75$ achieved better classification performance than other α values. Therefore, $\alpha = 0.75$ is used in the experiments in PSOPRSN and MOPSOPRS.

5.2 Experimental Results of PSOPRSN and MOPSOPRS

PSOPRSN obtains a single solution in each of the 30 independent runs. MOPSOPRS obtains a set of non-dominated solutions in each run. To compare these two kinds of results, the 30 solutions (from 30 runs) resulted from PSOPRSN are presented in this section. 30 sets of feature subsets achieved by MOPSOPRS are firstly combined into one union set. In the union set, for the feature subsets including the same number of features (e.g. m), their classification error rates are averaged. Therefore, a set of average solutions is obtained by using the average classification error rates and the corresponding numbers (e.g. m). The set of average solutions is called the *average* Pareto front and presented here. Meanwhile, the non-dominated solutions in the union set are called the *best* Pareto front and are also presented to compare with the solutions achieved by PSOPRSN.

Figure 1 shows the experimental results of MOPSOPRS and PSOPRSN with $\gamma = 0.5$ and $\gamma = 0.9$ on the test sets, where DT was used as the classification algorithm. In the figure, each chart corresponds to one of the dataset used in the experiments. On the top of each chart, the numbers in the brackets show the number of 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. As the results of using NB and 5NN are similar to that of using DT, the results of using NB and 5NN are not presented here due the page limit.

In Figure 1, "AvePar" and "BestPar" stand for the *average* and the *best* Pareto fronts resulted from MOPSOPRS in the 30 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, PSOPRSN may evolve the same feature subset in different runs and they are shown in the same point in the chart. Therefore, although 30 results are presented in $\gamma = 0.5$ and $\gamma = 0.9$, there may be less than 30 distinct points shown in one chart.

Results of PSOPRSN. As can be seen from Figure 1, in most cases, PSOPRSN with both $\gamma = 0.5$ and $\gamma = 0.9$ selected feature subsets with a smaller

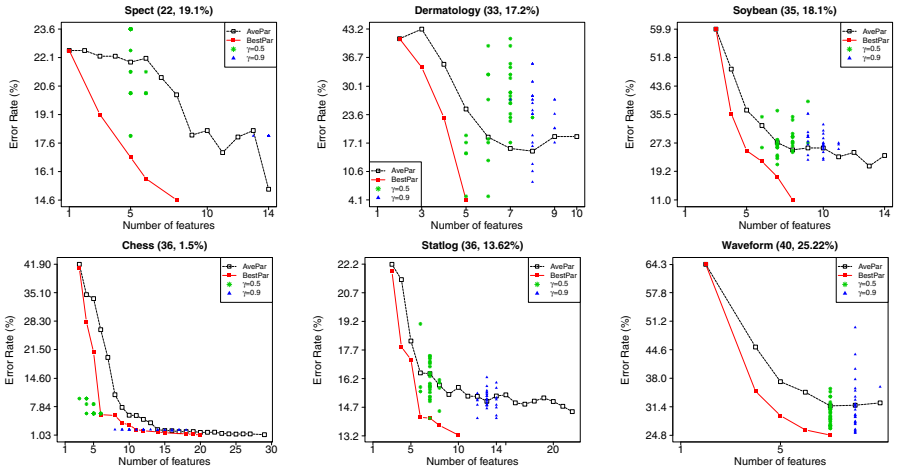

Fig. 1. Results of MOPSOPRS and PSOPRSN on test sets evaluated by DT

number of features and achieved similar or even better classification performance than using all features. $\gamma = 0.9$ achieved similar or better classification than $\gamma = 0.5$ and $\gamma = 0.5$ usually achieved a smaller number of features than $\gamma = 0.9$. The reason is that the number of features is assigned more important in $\gamma = 0.5$ than in $\gamma = 0.9$.

Results of MOPSOPRS. According to Figure 1, in three of the six datasets, the average Pareto front of MOPSOPRS (AvePar) includes two or more solutions, which selected a smaller number of features and achieved a similar or even lower classification error rate than using all features. For the same number of features, there are a variety of combinations of features with different classification performances. The feature subsets obtained in different runs may include the same number of features but different classification error rates. Therefore, although the solutions obtained in each run are non-dominated, some solutions in the average Pareto front may dominate others. This also happens when using 5NN or NB as the classification algorithms.

According to Figure 1, in *all* datasets, the non-dominated solutions in MOPSOPRS (BestPar) selected one or more feature subsets, which included less than one third of features and achieved better classification performance than using all features.

Comparisons Between MOPSOPRS and PSOPRSN. In most datasets, solutions in AvePar achieved similar results to both $\gamma = 0.5$ and $\gamma = 0.9$ in terms of both 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 with a smaller number of features than both $\gamma = 0.5$ and $\gamma = 0.9$, especially in the datasets with a large number of features, such as the Statlog and Waveform datasets.

Figure 1 shows that MOPSOPRS can further reduce the number of features and increase the classification performance, which indicates that MOPSOPRS

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

Dataset	Spect		Dermatology		Soybean		Chess		Statlog		Waveform	
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

as a multi-objective technique can explore the search space of a feature selection problem better than the single objective algorithm, PSOPRSN.

Note that the results of using the three classification algorithms (DT, NB and 5NN) show that the performance of MOPSOPRS and PSOPRSN are consistent when using different classification algorithms, which suggests the proposed filter algorithm with probabilistic rough set as the evaluation criterion are general to these three classification algorithms.

5.3 Comparisons with Two Traditional Algorithms

Table 3 shows the results of CfsF and CfsB for feature selection, where DT was used for classification. Comparing MOPSOPRS (results of using DT shown in Figure 1) with CfsF and CfsB, it can be seen that in *all* datasets, MOPSOPRS (BestPar) outpormed both CfsF and CfsB in terms of the classification performance and the number of features.

6 Conclusions

This work conducted the first study on PSO and rough set theory for multi-objective feature selection. We proposed a novel feature algorithm (MOPSOPRS) based on a multi-objective PSO and probabilistic rough set theory with the goal of obtaining a set of non-dominated feature subsets, which reduced the number of features and achieved similar or even better classification performance than all features. MOPSOPRS was examined and compared with three single objective algorithms (PSOPRS, PSOPRS and PSOPRSN) and two traditional methods on six datasets of varying difficulty. DT NB and 5NN were used to test the generality of MOPSOPRS. Experimental results show that in almost all cases, MOPSOPRS can automatically evolve a set of non-dominated feature subsets that include a smaller number of features and achieve better classification performance (evaluated by the three classification methods) than using all features. MOPSOPRS outperformed the three PSO based single objective and the two traditional algorithms in terms of both the number of features and the classification performance. The results also show that MOPSOPRS are general to the three different classification algorithms. This study finds that as a multi-objective algorithm, MOPSOPRS can search the solution space effectively to obtain a set of non-dominated solutions instead of a single solution. Examining the Pareto front achieved by the multi-objective algorithm can assist users in choosing their preferred solutions to meet their own requirements.

In future, we will further investigate the use of multi-objective PSO and probabilistic rough set for feature selection to better explore the Pareto front of non-dominated solutions in feature selection problems.

References

1. Almuallim, H., Dietterich, T.G.: Learning boolean concepts in the presence of many irrelevant features. Artificial Intelligence 69, 279–305 (1994)
2. Cervante, L., Xue, B., Shang, L., Zhang, M.: A Dimension Reduction Approach to Classification Based on Particle Swarm Optimisation and Rough Set Theory. In: Thielscher, M., Zhang, D. (eds.) AI 2012. LNCS, vol. 7691, pp. 313–325. Springer, Heidelberg (2012)
3. Chakraborty, B.: Genetic algorithm with fuzzy fitness function for feature selection. In: International Symposium on Industrial Electronics, vol. 1, pp. 315–319 (2002)
4. Dash, M., Liu, H.: Feature selection for classification. Intelligent Data Analysis 1(4), 131–156 (1997)
5. Frank, A., Asuncion, A.: UCI machine learning repository (2010)
6. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. The Journal of Machine Learning Research 3, 1157–1182 (2003)
7. Hall, M.A.: Correlation-based Feature Subset Selection for Machine Learning. Ph.D. thesis, The University of Waikato, Hamilton, New Zealand (1999)
8. Jensen, R.: Performing Feature Selection with ACO. In: Abraham, A., Grosan, C., Ramos, V. (eds.) Swarm Intelligence in Data Mining. SCI, vol. 34, pp. 45–73. Springer, Heidelberg (2006)
9. Kennedy, J., Eberhart, R.: A discrete binary version of the particle swarm algorithm. In: IEEE International Conference on Systems, Man, and Cybernetics, vol. 5, pp. 4104–4108 (1997)
10. Marill, T., Green, D.: On the effectiveness of receptors in recognition systems. IEEE Transactions on Information Theory 9(1), 11–17 (1963)
11. Mohemmed, A., Zhang, M., Johnston, M.: Particle swarm optimization based adaboost for face detection. In: IEEE Congress on Evolutionary Computation (CEC 2009), pp. 2494–2501 (2009)
12. Neshatian, K., Zhang, M.: Pareto front feature selection: using genetic programming to explore feature space. In: Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation, New York, NY, USA, pp. 1027–1034 (2009)
13. Pawlak, Z.: Rough sets. International Journal of Parallel Programming 11, 341–356 (1982)
14. Shi, Y., Eberhart, R.: A modified particle swarm optimizer. In: IEEE International Conference on Evolutionary Computation (CEC 1998), pp. 69–73 (1998)
15. Sierra, M.R., Coello Coello, C.A.: Improving PSO-Based Multi-objective Optimization Using Crowding, Mutation and ϵ-Dominance. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 505–519. Springer, Heidelberg (2005)
16. Stearns, S.: On selecting features for pattern classifier. In: Proceedings of the 3rd International Conference on Pattern Recognition, pp. 71–75 (1976)
17. Unler, A., Murat, A.: A discrete particle swarm optimization method for feature selection in binary classification problems. European Journal of Operational Research 206(3), 528–539 (2010)
18. Wang, X., Yang, J., Teng, X., Xia, W., Jensen, R.: Feature selection based on rough sets and particle swarm optimization. Pattern Recognition Letters 28(4), 459–471 (2007)
19. Whitney, A.: A direct method of nonparametric measurement selection. IEEE Transactions on Computers C-20(9), 1100–1103 (1971)
20. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 2nd edn. Morgan Kaufmann (2005)
21. Yao, Y., Zhao, Y.: Attribute reduction in decision-theoretic rough set models. Information Sciences 178(17), 3356–3373 (2008)