Feature Selection Based on PSO and Decision-Theoretic Rough Set Model

Aneta Stevanovic, Bing Xue, Mengjie Zhang
School of Engineering and Computer Science
Victoria University of Wellington, PO Box 600, Wellington, New Zealand
{Aneta.Stevanovic, Bing.Xue, Mengjie.Zhang}@ecs.vuw.ac.nz

***Abstract*—In this paper, we propose two new methods for feature selection based on particle swarm optimisation and a probabilistic rough set model called decision-theoretic rough set (DTRS). The first method uses rule degradation and cost properties of DTRS in the fitness function. This method focuses on the quality of the selected feature subset as a whole. The second method extends the first one by adding the individual feature confidence to the fitness function, which measures the quality of each feature in the subset. Three learning algorithms are employed to evaluate the classification performance of the proposed methods. The experiments are run on six commonly used datasets of varying difficulty. The results show that both methods can achieve good feature reduction rates with similar or better classification performance. Both methods can outperform two traditional feature selection methods. The second proposed method outperforms the first one in terms of the feature reduction rates while being able to maintaining similar or better classification rates.**

I. Introduction

Reducing the number of features in a dataset is an important step in many classification problems. Datasets often include a large number of features in an attempt to describe its domain as well as possible. This can lead to the creation of many irrelevant and redundant features which impairs the classification process. This problem can be avoided by using feature selection whose goal is to find a subset of features to achieve similar or better classification accuracy than using all features. Feature selection can decrease computational time and provide more general view of the underlying relationship in datasets [1].

There exist two broad feature selection approaches: wrapper and filter approaches. Their difference is in the evaluation measures of the selected feature subsets. Wrappers include a classification/learning algorithm to evaluate the goodness of the selected features. They can achieve better classification accuracy than filter approaches. However, the disadvantages are their high computational costs because each evaluation requires the classification algorithm to be run, and loss of generality [2]. Filters select features based on the inherent properties of a dataset. They are independent of a classification algorithm and as such tend to be more efficient and general [3]. However, the design of evaluation criteria is not as straightforward as in wrappers. Many different criteria can be used, including distance, dependency, and consistency measures [1].

Since the search space grows exponentially as the number of features increases, performing an exhaustive search is not practical [2]. As a result, the choice of a search method is important. Various greedy algorithms have been used for feature selection [1]. However, many have trouble finding solutions beyond a local optima [4]. Evolutionary computation (EC) techniques are well-known for their global search ability and can be a candidate technique for addressing this problem.

One such EC technique is particle swarm optimisation (PSO) [5]. Compared with genetic programming and genetic algorithms, PSO is easier to implement, computationally less expensive, and can converge more quickly [6]. Previous research has shown that PSO can be successfully applied to feature selection problems [4], [7], [8], [9], [10].

Rough set theory developed by Pawlak [11] is a mathematical approach to imperfect knowledge. It can be used as an evaluation criteria in a filter based feature selection approach. Wang et al. [12] achieved good results using PSO based on rough set theory to find optimal feature subsets. However, rough set theory applied to feature selection has some limitations. For example, the uncertainty of the boundary region is not considered. Probabilistic rough set models, one of which is decision-theoretic rough set (DTRS) model, can address such limitations with the introduction of probabilistic threshold values into the standard model along with several different evaluation properties, such as decision-monotocity, confidence and cost [13]. However, these evaluation properties have not been applied to feature selection together with PSO.

A. Goals

The overall goal is to develop a PSO based filter approach to feature selection problems using the DTRS model as the evaluation measure. The expectation is that this approach is able to achieve reduction in the number of features while maintaining or improving the classification accuracy over using all original features. To achieve this goal, two new methods are developed by using different evaluation properties of DTRS in the fitness function. The new algorithms will be validated using six commonly used UCI datasets [14] of varying difficulty and compared with three existing PSO and rough set based algorithms and two traditional feature selection methods. Specifically, we will investigate:

- whether the first algorithm which considers the preservation of decision rules and their cost can reduce the number of features while maintaining good classification accuracy;

- whether the second algorithm which besides the preservation of decision rules and their cost also considers the individual confidences of selected features can further reduce the number of features and maintain or improve the classification accuracy;
- whether the two proposed methods can outperform the three PSO based algorithms and the two traditional feature selection methods.
- whether the proposed filter algorithms are general to different classification algorithms.

B. Organisation

The rest of the paper is organised as follows: Background information and related work are presented in section II. Proposed methods are detailed in section III. Sections IV and V contain the description of experimental design and the discussions on the experimental results, respectively. Finally, section VI provides conclusions.

II. Background

A. Particle Swarm Optimisation (PSO)

PSO is an evolutionary computation search technique developed by Kennedy and Eberhart [5], which stimulates social behaviours such as bird flocking and fish schooling. The search is performed using a population (called swarm) of particles, where each particle represents a candidate solution. A particle has a position vector and a velocity vector denoted by $x_i = (x_{i1}, x_{i2}, ..., x_{iD})$ and $v_i = (v_{i1}, v_{i2}, ..., v_{iD})$, respectively, where D is the dimension and i is the index of a particle. In each iteration, a particle's vectors are updated based on its past personal experience as well as the past experience of the whole swarm. More specifically, each particle remembers its personal best visited position (*pbest*) and the swarm records the best position obtained by any particle in the population so far (*gbest*). The following equations shows how these two values are used to move the particles through the search space:

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

$$x_{id}^{t+1} = x_{id}^{t} + v_{id}^{t+1} \quad (2)$$

where *t* is the *t*th PSO iteration, $d \in D$ is the dth dimension of a particle, w is inertia weight, c_1 and c_2 are acceleration constants, and r_1 and r_2 are random numbers with uniform distribution in [0, 1].

Many optimisation problems, of which feature selection is one, are set in a discrete space. For this reason, Kenedy and Eberhart [15] proposed a binary PSO (BPSO) where every dimension of a particle's position is restricted to zero or one. The velocity must be transformed to [0, 1] interval which is achieved by using a sigmoid function $s(v_{id})$. These changes give the following rule for updating a particle position:

$$x_{id} = \begin{cases} 1, & \text{if } rand() < s(v_{id}) \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where

$$s(v_{id}) = \frac{1}{1 + e^{-v_{id}}} \quad (4)$$

and $rand()$ is a random number with uniform distribution in [0, 1].

B. Rough Set Theory

Rough set theory [11] is a mathematical approach to imperfect knowledge. It can deal with data uncertainty and vagueness without any additional information about the data.

Information and knowledge about the data can be represented by an information table $I = (U, A)$, where U is a finite set of objects and A is a finite set of attributes that describe the objects. An equivalence relation with respect to a reduct $R \subseteq A$ is defined as $IND(R) = \{(x, y) \in U^2 | \forall a \in R, a(x) = a(y)\}$. The two objects in U satisfy the relation if and only if they have the same values on all attributes in R, meaning they are indiscernible from each other. The equivalence relation $IND(R)$ partitions U (U/R) into equivalence classes denoted as $[x]_R$.

For a subset $X \subseteq U$, the lower approximation ($\underline{R}X$) and upper approximation ($\overline{R}X$) of X with respect to the partition U/R is defined as

$$\underline{R}X = \{x \in U | [x]_R \subseteq X\} = \{x \in U | P(X|[x]) = 1\} \quad (5)$$

$$\overline{R}X = \{x \in U | [x]_R \cap X \neq \emptyset\} = \{x \in U | P(X|[x]) > 0\} \quad (6)$$

where $P(X|[x]) = \frac{|[x]_R \cap X|}{|[x]_R|}$ is the conditional probability that an object x in the equivalence class $[x]_R$ belongs to X.

Based on $\underline{R}X$ and $\overline{R}X$, the universe U can be separated into three disjoint regions: the positive region (POS(X) = $\underline{R}X$) containing equivalence classes which can with certainty induce the decision class of X, the boundary region (BND(X) = $\overline{R}X$ - $\underline{R}X$) whose equivalence classes can induce a partial decision of X, and the negative region (NEG(X) = $U - \overline{R}X$) which includes all equivalence classes that for sure cannot induce the decision class of X.

The lower approximation ($\underline{R}X$) in standard rough set theory with its condition $P(X|[x]) = 1$ requires that an equivalence class is completely contained within X, which is often too restrictive in classification problems. Also, the equivalence classes in the upper approximation must have a non-empty intersection with X which ignores the uncertainty of the boundary region. Decision-theoretic rough set (DTRS) model, which is a probabilistic rough set model [16], introduces a pair of probabilistic thresholds $\alpha, \beta \in [0, 1]$ with $\alpha \geq \beta$ which are used to relax the approximation criteria. Consequently, the approximation equations are modified into $\underline{R}X = \{x \in U | P(X|[x]) \geq \alpha\}$ and $\overline{R}X = \{x \in U | P(X|[x]) > \beta\}$. The condition $\alpha \geq \beta$ ensures that the lower approximation is smaller than the upper approximation. Also, the α value should be greater than 0.5 to make sure that the positive region is the dominant one.

C. Recent Work on Feature Selection

1) EC Algorithms for Feature Selection: It has been shown that EC algorithms can give good results when applied to the feature selection problem [17], [18].

AlSukker et al. [19] propose a method to overcome premature convergence that genetic algorithms (GAs) suffer from. The proposed method (DGA) is a GA based on enhanced population diversity, parents' selection and improved genetic operators. The results show that DGA outperformed other GAs with a similar computation cost. However, its limitation is that its performance suffers when applied to high redundant data with high dimensionality.

Huang et al. [20] propose a hybrid GA for feature selection, which consists of two optimisation stages. The outer optimisation stage performs the global search for the best subset of features in a wrapper way, while the inner optimisation performs the local search in a filter manner in which the redundancy of the already selected features is taken into account. The experimental results show both good feature reduction and classification rates. A drawback of this method is its long running time. Babaoglu et al. [21] have done a comparison of feature selection models based on GA and BPSO on coronary artery disease data. Classification performance and efficiency are compared and the results show that BPSO is more successful than GA.

Neshatian and Zhang [22] propose a genetic programming (GP) approach where the building blocks are subsets of features and set operators. GP combines these subsets and set operators to find an optimal subset of features. The experiments performed on highly imbalanced face detection problems demonstrate that this approach is effective in terms of the dimensionality reduction and processing time.

Another GP approach proposed by Neshatian and Zhang [23] turns to feature scoring as a way of evaluating selected feature subsets. The features are assigned a score in the context of other features participating in a GP program. The results show that the proposed feature ranking method can identify important features and obtain the same classification performance as when all features are used.

2) PSO for Feature Selection: PSO based feature selection methods gained attention in recent years. Wang et al. [12] propose a feature selection approach based on rough sets and PSO. The rough set degree of dependency is used to evaluate the fitness of each particle. They show that compared with GAs, PSO is computationally inexpensive in terms of both memory and runtime and at the same time can give promising feature reduction results.

Unler and Murat [24] develop a modified BPSO algorithm for the feature selection. This approach dynamically accounts for the relevance and dependence of any features to be added to the already selected ones. Experiments suggest that the proposed method outperforms the tabu search and scatter search algorithms.

Abdul-Rahman et al. [9] propose a new strategy based on PSO and rough set theory (RST). It has characteristics of both wrapper and filter approaches. RST and its discernibility relation is used to pre-reduce the feature set before optimisation by PSO. Experimental results show that the proposed method significantly improves classification and feature reduction rates for most datasets.

Two different approaches based on BPSO are proposed by Cervante et al. [10], [25]. The first employs DTRS and the second one employs mutual information and entropy to evaluate the selected feature subsets. The results from both approaches show that with proper weights they can usually select a smaller feature subset with similar or better classification rates.

Previous research has shown that PSO and rough set theory are effective techniques for feature selection problems. Little work has been done on DTRS which was argued to be a good evaluation measure [13]. The work has mostly focused on the fitness functions based on the positive probabilistic region in DTRS, while the effect of boundary region has been taken into account.

III. Proposed Methods

In this section, we propose two new methods based on DTRS model [13] which calculates the probabilistic thresholds (α, β) based on a set of loss functions. The loss functions represent costs of decision rules, which are derived from the concept of three regions in rough set theory.

In DTRS, a decision rule $[x] \to D_i$ indicates that an object with description $[x]$ would be in the decision class D_i. If $[x] \subseteq \mathrm{POS}_{\alpha,\beta}(U_i)$, a positive rule is induced ($[x] \to_P U_i$). Otherwise, if $[x] \subseteq \mathrm{BND}_{\alpha,\beta}(U_i)$, a boundary rule is induced ($[x] \to_B U_i$), or if $[x] \subseteq \mathrm{NEG}_{\alpha,\beta}(U_i)$, a negative rule is induced ($[x] \to_N U_i$).

Confidence of a decision rule in DTRS is a quantitative measure, which is defined as the ration of the number of objects in an equivalence class $[x]$ that are correctly classified as the decision class D_i and the number of objects in the equivalence class $[x]$:

$$confidence([x] \to D_i) = \frac{|[x] \cap D_i|}{|[x]|} \tag{7}$$

where $|.|$ is the set cardinality. The higher the confidence value, the more valuable the rule is.

Corresponding to the previously described three decision rules, three actions a_P, a_N and a_B can be taken when deciding an object to be in POS(X), NEG(X) or BND(X) region, respectively. Each action comes with a cost or risk value. If an object belongs to X, then λ_{PP}, λ_{BP} and λ_{NP} specify the costs of taking the actions a_P, a_N and a_B. If an object does not belong to X, then λ_{PN}, λ_{BN} and λ_{NN} are the costs of the actions. These costs have the following relationship with the probabilistic thresholds α and β in DTRS:

$$\begin{aligned} \alpha &= \tfrac{\lambda_{PN}-\lambda_{BN}}{(\lambda_{PN}-\lambda_{BN})+(\lambda_{BP}-\lambda_{PP})}, \\ \beta &= \tfrac{\lambda_{BN}-\lambda_{NN}}{(\lambda_{BN}-\lambda_{NN})+(\lambda_{NP}-\lambda_{BP})} \end{aligned} \tag{8}$$

A. Proposed Algorithm 1 (PSOCP)

A new algorithm PSOCP based on cost and preservation of decision rules is proposed. PSOCP aims to preserve the decision rules obtained from all features, as well as to minimise costs of deciding which region an object belongs to. These two criteria aims to ensure that the decrease of decision making cost does not worsen the induced decision rules for a selected subset of features [13].

Having the information table I from section II-B in mind, let $R \subseteq A$ be a reduct. For any object $x \in U$, the preservation of decision rules criterion can be represented by:

$$\begin{gathered} ([x]_A \to_P D_{max}([x]_A)) \Longrightarrow ([x]_R \to_P D_{max}([x])_A)), \text{and} \\ ([x]_A \to_B D_{max}([x]_A)) \Longrightarrow ([x]_R \to_{B/P} D_{max}([x])_A)) \end{gathered} \tag{9}$$

where D_{max} is a decision class for which equivalence class $[x]_A$ has the highest confidence. The criterion ensures that the positive and boundary rules obtained with all features do not degrade. In other words, for any x if $x \in POS_{U/A}(D_i)$, then $x \in POS_{U/R}(D_i)$. Similarly, if $x \in BND_{U/A}(D_i)$, then $x \in BND_{U/R}(D_i)$ or the rule is upgraded to $x \in POS_{U/R}(D_i)$.

The cost criterion is represented as the sum of costs of positive and boundary rule sets:

$$\Omega^{P \cup B}(U \to U/D) = \sum_{[x] \subseteq \text{POS}_{(\alpha,\beta)}(U/D)} \Omega(a_P|[x]) + \sum_{[x] \subseteq \text{BND}_{(\alpha,\beta)}(U/D)} \Omega(a_B|[x]) \quad (10)$$

where

$$\begin{array}{l} \Omega(a_P|[x]) = \lambda_{PP} P(X|[x]) \lambda_{PN} P(\neg X|[x]), \\ \Omega(a_B|[x]) = \lambda_{BP} P(X|[x]) \lambda_{BN} P(\neg X|[x]) \end{array} \quad (11)$$

The cost of the partition with respect to reduct R should not increase, meaning $\Omega(U/R \to U/D) \leq \Omega(U/A \to U/D)$.

The fitness function in PSOCP combines the preservation of decision rules and cost in the following manner:

$$Fitness_1(R) = ND + Cost \quad (12)$$

where ND is the number of non-degraded positive and boundary rules. $Cost$ is the difference between $(\lambda_{PP} + \lambda_{PN} + \lambda_{BN} + \lambda_{BP})$ and the value calculated by the Equation 10, which is a maximisation problem like ND. Both ND and $Cost$ are scaled to [0,1] so that neither value overwhelms the other.

B. Proposed Algorithm 2 (PSOCPC)

The fitness function $Fitness_1(R)$ aims to maximise the classification performance only. If several solutions have the same fitness value, but different numbers of features, $Fitness_1(R)$ does not prefer the solutions with a smaller number of selected features.

To address this issue, another new algorithm (PSOCPC) is proposed based on cost, preservation of decision rules and confidence, where $Fitness_2(R)$ is the fitness function. Besides the component which focuses on the classification performance (ND+$Cost$), $Fitness_2(R)$ includes the $AvgIC$ component which aims to reduce the number of features:

$$Fitness_2(R) = (1 - \gamma) * (ND + Cost) + \gamma * AvgIC \quad (13)$$

where $\gamma \in (0, 1)$ determines the relative importance of the two components. $AvgIC$ is the average of the individual confidences of the selected features. The individual confidence of a feature f from the selected features is calculated using the Equation 7. The equivalence class $[x]$ in the equation is obtained by partitioning the universe using the feature f.

There is expected to be some variations in the individual confidences of features and the $AvgIC$ component makes use of this. It tries to find a feature subset with as high average of individual confidences as possible. This can be achieved by removing features with poor individual confidences, which leads to decrease in the number of selected features. Similarly, the $AvgIC$ value is degraded in the solutions containing many low confidence features while the solutions without such features have higher $AvgIC$ value.

TABLE I
DATASETS

Dataset	#Features	#Classes	#Instances
Dermatology	33	6	366
Spect	22	2	267
LED Display	24	10	1000
Soybean (large)	35	19	683
Waveform	40	3	5000
Mushroom	22	2	5644

IV. Design of Experiments

Experiments have been conducted on six commonly used datasets from the UCI machine learning repository [14]. Their properties can be found in Table I. They are used as the representative of problems the proposed methods can address. All datasets contain categorical data and no missing values.

The datasets are randomly split into two parts with 2/3 of instances being training and 1/3 of instances being test set. As filter approaches, the proposed methods are run on the training set producing a subset of features. The classification accuracy of the subset is evaluated by learning algorithms on the unseen test set. Three learning algorithms are used to evaluate the generality of the proposed algorithms, which are decision tree (DT), naive Bayes (NB), and K-nearest neighbour (5-NN). The size of the swarm is 30 and the number of iterations is limited to 100, $w = 0.7298$, $c_1 = c_2 = 1.49618$. This setting is based on the experimental design described in [10], [26]. The results shown in next section are the testing classification rates over 30 independent runs.

In DTRS, different values are used for probabilistic thresholds α (0.9, 0.8, 0.7, 0.6) and β (0.1, 0.25, 0.4, 0.55, 0.65). All combinations have been tested with the exception of (α, β) = (0.6, 0.65) because of the $\alpha \geq \beta$ condition. The tests show the effect these threshold values have on the performance. The experiments on the PSOCPC algorithm are conducted with five different γ values (0.1, 0.3, 0.5, 0.7, 0.9), each over 30 independent runs, to see the effects of the variation in γ value on the performance.

The performance of the proposed methods are compared to the three existing feature selection methods based on PSO and rough set theory described in [10]. The three methods are called PSORS, PSOPRS and PSOPRSN. PSORS evaluates potential solutions using standard rough set theory. PSOPRS and PSOPRSN are both based on DTRS with one difference that the latter also considers the number of selected features in its fitness function. Two conventional filter feature selection methods (CfsF and CfsB) in Waikato Environment for Knowledge Analysis (Weka) [27] are also used for comparison purposes. CfsF and CfsB uses a correlation measures proposed by Hall [28] and employ forward and backward selection, respectively.

V. Results

Results are displayed in tables II, III, IV, V. “#Attr” is the average number of features selected by a proposed method

TABLE II
RESULTS OF THE PSOCP ALGORITHM WITH β = 0.25

	Dermatology			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	33	0.828	0.959	0.934
α				
0.6	18.3	0.848 (0.044, 0.943)	0.917 (0.035, 0.975)	0.897 (0.036, 0.975)
0.7	18.67	0.858 (0.040, 0.951)	0.924 (0.030, 0.967)	0.903 (0.033, 0.959)
0.8	18.5	0.853 (0.038, 0.943)	0.921 (0.033, 0.967)	0.898 (0.034, 0.959)
0.9	18.4	0.852 (0.040, 0.967)	0.917 (0.036, 0.967)	0.893 (0.033, 0.943)

	LED Display			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	24	1.0	1.0	0.895
α				
0.6	14.37	0.854 (0.157, 1.0)	0.850 (0.162, 1.0)	0.810 (0.168, 1.0)
0.7	14.23	0.845 (0.197, 1.0)	0.842 (0.197, 1.0)	0.805 (0.194, 1.0)
0.8	14.23	0.845 (0.197, 1.0)	0.842 (0.197, 1.0)	0.805 (0.194, 1.0)
0.9	14.27	0.824 (0.212, 1.0)	0.824 (0.207, 1.0)	0.785 (0.204, 1.0)

	Spect			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	22	0.809	0.764	0.786
α				
0.6	8.73	0.781 (0.028, 0.843)	0.770 (0.029, 0.843)	0.772 (0.027, 0.843)
0.7	15.3	0.808 (0.019, 0.843)	0.766 (0.019, 0.809)	0.781 (0.020, 0.809)
0.8	16.23	0.804 (0.019, 0.843)	0.773 (0.022, 0.832)	0.782 (0.014, 0.809)
0.9	16.43	0.812 (0.024, 0.843)	0.769 (0.023, 0.809)	0.786 (0.019, 0.820)

	Soybean			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	35	0.819	0.903	0.894
α				
0.6	20.77	0.814 (0.030, 0.877)	0.863 (0.028, 0.912)	0.830 (0.032, 0.894)
0.7	20.47	0.792 (0.040, 0.868)	0.843 (0.033, 0.912)	0.808 (0.042, 0.899)
0.8	20.47	0.792 (0.041, 0.868)	0.842 (0.038, 0.912)	0.802 (0.048, 0.899)
0.9	20.63	0.797 (0.035, 0.868)	0.845 (0.035, 0.912)	0.807 (0.049, 0.899)

	Waveform			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	40	0.748	0.797	0.801
α				
0.6	21.37	0.728 (0.027, 0.765)	0.758 (0.030, 0.798)	0.717 (0.040, 0.772)
0.7	21.37	0.728 (0.027, 0.765)	0.758 (0.030, 0.798)	0.717 (0.040, 0.772)
0.8	21.37	0.728 (0.027, 0.765)	0.758 (0.030, 0.798)	0.717 (0.040, 0.772)
0.9	21.37	0.728 (0.027, 0.765)	0.758 (0.030, 0.798)	0.717 (0.040, 0.772)

	Mashroom			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	22	1.0	0.961	1.0
α				
0.6	11.2	0.997 (0.003, 1.0)	0.946 (0.035, 0.986)	0.995 (0.004, 1.0)
0.7	11.63	0.999 (0.001, 1.0)	0.954 (0.030, 0.986)	0.997 (0.003, 1.0)
0.8	11.5	0.999 (0.001, 1.0)	0.956 (0.025, 0.986)	0.997 (0.003, 1.0)
0.9	11.73	0.999 (0.001, 1.0)	0.955 (0.002, 0.985)	0.951 (0.024, 1.0)

TABLE III
RESULTS OF THE PSOCP ALGORITHM WITH α = 0.8

	Dermatology			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	33	0.828	0.959	0.934
β				
0.1-0.55	18.5	0.883 (0.038, 0.943)	0.921 (0.033, 0.967)	0.898 (0.034, 0.959)
0.65	18.4	0.849 (0.041, 0.943)	0.919 (0.036, 0.967)	0.894 (0.037, 0.959)

	LED Display			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	24	1.0	1.0	0.895
β				
0.1-0.4	14.23	0.845 (0.197, 1.0)	0.842 (0.197, 1.0)	0.805 (0.194, 1.0)
0.55-0.65	14.2	0.845 (0.197, 1.0)	0.842 (0.197, 1.0)	0.805 (0.195, 1.0)

	Spect			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	22	0.809	0.764	0.786
β				
0.1-0.4	16.23	0.804 (0.019, 0.843)	0.773 (0.022, 0.832)	0.782 (0.014, 0.809)
0.55	16.07	0.805 (0.018, 0.843)	0.774 (0.020, 0.820)	0.792 (0.015, 0.820)
0.65	16.47	0.808 (0.018, 0.843)	0.774 (0.020, 0.820)	0.777 (0.014, 0.809)

	Soybean			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	35	0.819	0.903	0.894
β				
0.1-0.4	20.47	0.792 (0.041, 0.868)	0.842 (0.038, 0.912)	0.802 (0.048, 0.899)
0.55	20.6	0.788 (0.036, 0.868)	0.848 (0.030, 0.912)	0.810 (0.035, 0.877)
0.65	20.73	0.784 (0.036, 0.863)	0.839 (0.037, 0.890)	0.797 (0.051, 0.863)

	Waveform			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	40	0.748	0.797	0.801
β				
0.1-0.65	21.37	0.728 (0.027, 0.765)	0.758 (0.030, 0.798)	0.717 (0.040, 0.772)

	Mushroom			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	22	1.0	0.961	1.0
β				
0.1-0.55	11.5	0.999 (0.001, 1.0)	0.956 (0.025, 0.956)	0.997 (0.003, 1.0)
0.65	11.47	0.999 (0.001, 1.0)	0.956 (0.025, 0.986)	0.997 (0.003, 1.0)

over 30 independent runs. The average test classification accuracy (Ave), the standard deviation (Sd), and the best classification accuracy (Best), are shown for each learning algorithm ("DT", "NB" and "NN").

A. Results of the PSOCP Algorithm

The results in Tables II and III show that PSOCP can significantly reduce the number of features while maintaining similar or better classification rates in most cases.

Table II displays the results of different α values when $\beta = 0.25$. The α value indicates how tolerant the fitness function is in accepting reducts to the positive rough set region. It can be seen that the classification rates are competitive to using all features in most datasets. The only exception is LED Display which has a perfect classification rate using all features. The classification rates of the DT and NB learning algorithms are improved in the Dermatology and Spect datasets, respectively. The classification rates and their standard deviation are similar to each other across different α values. The best rates achieved by the selected features are equal to or greater than the ones achieved by all features in all cases.

The proposed algorithm can achieve the reduction rates of at least 25%. However, the change in α value does not bring about notable change in reduction rates. The only exception is Spect dataset which can be explained by the fact that the fitness function does not have a mechanism to give preference to smaller reducts. That is, as long as two different sized reducts preserve the decision rules and their cost, the function treats these reducts the same. The exception in Spect dataset suggests that the dataset contains more redundant features. Consequently, it is more common for smaller sized reducts to be more valuable as they are less likely to have as many redundant features as larger reducts.

Table III displays what kind of effect different β values have on the performance of the PSOCP algorithm. The larger the β value is, the smaller boundary rough set region becomes. Smaller boundary regions in theory reduce the uncertainty of how to classify instances, but at the same time can introduce more classification errors. The results indicate that the β value has little to no effect. Both reduction and classification rates change only slightly for larger values of β and this appears to be dataset dependent. For example, the number of features is the smallest in Dermatology, LED Display and Mushroom datasets when $\beta = 0.65$, while not for other datasets.

B. Results of the PSOCPC Algorithm

According to the results in Table IV and V, the PSOCPC algorithm can further reduce the number of features while mostly achieving similar or better classification rates.

From Table IV, it can be seen that the classification rates of Spect and Soybean datasets are the best when the smallest number of features is selected. Moreover, LED Display dataset can achieve perfect or close to perfect classification rates with a significantly reduced number of features. This indicates that the dataset contains a lot of noisy data. Once the features containing this data are removed, classification rate can come very close to perfect. The obtained best classification rates with selected features are similar or better than those obtained with all features with the exception of Soybean.

In all datasets, the average number of selected features decreases with the decrease of α value. Slight deviation from this pattern is present in Mashroom dataset with lower α values. The acceptance of instances into the positive region is more lenient with smaller α values, which develops the ability to remove additional redundant features from reducts. The most dramatic reduction across all α values can be seen in Mushroom. The two datasets with most features, Dermatology and Soybean, see the feature reduction rates of at least 42%.

Similarly to the PSOCP algorithm, different β values do not have noticeable effect on the algorithm performance and for that reason these experimental results are not shown. This suggests that the boundary rough set region plays a minor role in performance of the proposed algorithms.

Table V shows the performance of PSOCPC with different weights applied to its two terms. It can been seen that in most cases the number of features is the lowest when the highest importance, i.e. largest γ value, is placed on the average individual confidence component. This is consistent with the previous expectation that the addition of the individual confidence to $Fitness_1(R)$ would further reduce the number of features.

C. PSOCP VS PSOCPC

When comparing classification rates, the PSOCPC algorithm generally achieves better results. This is most obvious in LED Display dataset where the average classification rates are greatly improved. In addition, the performance in Spect dataset when $\alpha = 0.6$ is significantly better. Moreover, most standard deviations of the classifiers in PSOCPC are smaller than the PSOCP algorithm. This shows that PSOCPC is more stable and consistent. The PSOCP algorithm has the advantage with NB and NN learning algorithms in Dermatology dataset where both average classification rates and standard deviations are improved. However, these rates are achieved using greater number of selected features than in PSOCPC.

TABLE IV
RESULTS OF THE PSOCPC ALGORITHM WITH β = 0.25 AND γ = 0.5

		Dermatology		
	#Attr	DT Ave (Sd, Best)	NB Ave (Sd, Best)	NN Ave (Sd, Best)
All	33	0.828	0.959	0.934
α				
0.6	11.27	0.850 (0.031, 0.943)	0.815 (0.038, 0.893)	0.844 (0.036, 0.959)
0.7	11.73	0.860 (0.024, 0.943)	0.829 (0.046, 0.967)	0.851 (0.031, 0.951)
0.8	12.83	0.873 (0.042, 0.967)	0.848 (0.060, 0.992)	0.875 (0.051, 0.975)
0.9	17.53	0.896 (0.042, 0.951)	0.912 (0.041, 0.967)	0.928 (0.039, 0.992)
		LED Display		
	#Attr	DT Ave (Sd, Best)	NB Ave (Sd, Best)	NN Ave (Sd, Best)
All	24	1.0	1.0	0.895
α				
0.6	6.57	0.988 (0.065, 1.0)	0.988 (0.066, 1.0)	0.988 (0.065, 1.0)
0.7	6.57	1.0 (0.0, 1.0)	1.0 (0.0, 1.0)	1.0 (0.0, 1.0)
0.8	6.57	1.0 (0.0, 1.0)	1.0 (0.0, 1.0)	1.0 (0.0, 1.0)
0.9	6.73	0.991 (0.047, 1.0)	0.991 (0.047, 1.0)	0.990 (0.052, 1.0)
		Spect		
	#Attr	DT Ave (Sd, Best)	NB Ave (Sd, Best)	NN Ave (Sd, Best)
All	22	0.809	0.764	0.786
α				
0.6	2.8	0.843 (0.0, 0.843)	0.838 (0.018, 0.843)	0.837 (0.022, 0.847)
0.7	12.33	0.796 (0.017, 0.843)	0.771 (0.026, 0.809)	0.778 (0.018, 0.809)
0.8	15.27	0.803 (0.016, 0.832)	0.786 (0.015, 0.809)	0.788 (0.016, 0.843)
0.9	15.3	0.802 (0.019, 0.832)	0.771 (0.019, 0.820)	0.784 (0.022, 0.832)
		Soybean		
	#Attr	DT Ave (Sd, Best)	NB Ave (Sd, Best)	NN Ave (Sd, Best)
All	35	0.819	0.903	0.894
α				
0.6	16.9	0.819 (0.028, 0.881)	0.840 (0.037, 0.899)	0.833 (0.046, 0.921)
0.7	19.57	0.796 (0.018, 0.828)	0.835 (0.021, 0.868)	0.811 (0.032, 0.859)
0.8	20.1	0.802 (0.024, 0.850)	0.838 (0.025, 0.894)	0.819 (0.033, 0.886)
0.9	19.73	0.796 (0.029, 0.850)	0.834 (0.027, 0.877)	0.820 (0.030, 0.863)
		Waveform		
	#Attr	DT Ave (Sd, Best)	NB Ave (Sd, Best)	NN Ave (Sd, Best)
All	40	0.748	0.797	0.801
α				
0.6	13.6	0.757 (0.021, 0.782)	0.781 (0.025, 0.817)	0.788 (0.029, 0.817)
0.7	13.6	0.757 (0.021, 0.782)	0.781 (0.025, 0.817)	0.788 (0.029, 0.817)
0.8	13.53	0.757 (0.021, 0.782)	0.780 (0.025, 0.817)	0.788 (0.028, 0.816)
0.9	13.53	0.757 (0.021, 0.782)	0.780 (0.025, 0.817)	0.788 (0.028, 0.816)
		Mashroom		
	#Attr	DT Ave (Sd, Best)	NB Ave (Sd, Best)	NN Ave (Sd, Best)
All	22	1.0	0.961	1.0
α				
0.6	1.87	0.985 (0.006, 0.994)	0.984 (0.006, 0.993)	0.985 (0.006, 0.994)
0.7	1.57	0.984 (0.006, 0.994)	0.983 (0.006, 0.993)	0.984 (0.006, 0.994)
0.8	2.6	0.986 (0.007, 1.0)	0.980 (0.010, 0.993)	0.986 (0.007, 1.0)
0.9	4.33	0.997 (0.005, 1.0)	0.968 (0.020, 0.993)	0.997 (0.005, 1.0)

The PSOCPC algorithm also achieves better results in feature reduction rates regardless of what the α value is. The number of features in Mushroom and LED Display datasets are more than halved compared with PSOCP performance. In the other datasets, the number of features is considerably reduced, especially when using lower α values.

TABLE V
RESULTS OF THE PSOCPC ALGORITHM WITH (α,β) = (0.7, 0.25)

	Dermatology			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	33	0.828	0.959	0.934
γ				
0.9	11.27	0.856 (0.025, 0.943)	0.819 (0.038, 0.918)	0.849 (0.027, 0.926)
0.7	11.83	0.854 (0.027, 0.943)	0.823 (0.039, 0.893)	0.848 (0.026, 0.951)
0.5	11.73	0.860 (0.024, 0.943)	0.829 (0.046, 0.967)	0.851 (0.031, 0.951)
0.3	12	0.861 (0.031, 0.943)	0.831 (0.046, 0.967)	0.852 (0.032, 0.951)
0.1	12	0.861 (0.031, 0.943)	0.831 (0.046, 0.967)	0.852 (0.032, 0.951)

	LED Display			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	24	1.0	1.0	0.895
γ				
0.9	6.7	0.983 (0.076, 1.0)	0.983 (0.075, 1.0)	0.983 (0.076, 1.0)
0.7	6.77	0.991 (0.047, 1.0)	0.991 (0.047, 1.0)	0.990 (0.052, 1.0)
0.5	6.57	1.0 (0.0, 1.0)	1.0 (0.0, 1.0)	1.0 (0.0, 1.0)
0.3	6.53	1.0 (0.0, 1.0)	1.0 (0.0, 1.0)	1.0 (0.0, 1.0)
0.1	6.6	1.0 (0.0, 1.0)	1.0 (0.0, 1.0)	1.0 (0.0, 1.0)

	Spect			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	22	0.809	0.764	0.786
γ				
0.9	10.17	0.791 (0.024, 0.843)	0.752 (0.027, 0.843)	0.769 (0.022, 0.843)
0.7	12.53	0.795 (0.016, 0.843)	0.764 (0.026, 0.798)	0.775 (0.018, 0.809)
0.5	12.33	0.796 (0.017, 0.843)	0.771 (0.026, 0.809)	0.778 (0.018, 0.809)
0.3	12.07	0.796 (0.019, 0.843)	0.764 (0.026, 0.809)	0.777 (0.018, 0.809)
0.1	11.97	0.799 (0.018, 0.843)	0.764 (0.025, 0.809)	0.772 (0.017, 0.809)

	Soybean			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	35	0.819	0.903	0.894
γ				
0.9	18.43	0.824 (0.030, 0.872)	0.853 (0.026, 0.890)	0.851 (0.035, 0.916)
0.7	19.2	0.803 (0.029, 0.863)	0.834 (0.032, 0.894)	0.810 (0.044, 0.934)
0.5	19.57	0.796 (0.018, 0.828)	0.835 (0.021, 0.868)	0.811 (0.032, 0.859)
0.3	19.5	0.801 (0.019, 0.833)	0.829 (0.026, 0.868)	0.814 (0.031, 0.859)
0.1	19.37	0.798 (0.020, 0.833)	0.829 (0.028, 0.859)	0.816 (0.034, 0.912)

	Waveform			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	40	0.748	0.797	0.801
γ				
0.9	13.37	0.755 (0.020, 0.782)	0.779 (0.025, 0.817)	0.787 (0.028, 0.801)
0.7	13.43	0.755 (0.021, 0.782)	0.781 (0.025, 0.817)	0.788 (0.028, 0.816)
0.5	13.6	0.757 (0.021, 0.782)	0.781 (0.025, 0.817)	0.788 (0.029, 0.817)
0.3	13.47	0.757 (0.021, 0.782)	0.780 (0.025, 0.817)	0.788 (0.029, 0.817)
0.1	13.57	0.757 (0.020, 0.782)	0.781 (0.025, 0.817)	0.789 (0.028, 0.817)

	Mashroom			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
All	22	1.0	0.961	1.0
γ				
0.9	1.73	0.984 (0.006, 0.997)	0.983 (0.006, 0.997)	0.984 (0.006, 0.997)
0.7	1.6	0.984 (0.006, 0.994)	0.984 (0.006, 0.993)	0.984 (0.006, 0.994)
0.5	1.57	0.984 (0.006, 0.994)	0.983 (0.006, 0.993)	0.984 (0.006, 0.994)
0.3	1.6	0.984 (0.006, 0.994)	0.983 (0.006, 0.993)	0.984 (0.006, 0.994)
0.1	1.6	0.984 (0.006, 0.994)	0.983 (0.006, 0.993)	0.984 (0.006, 0.994)

D. Comparisons with Existing PSO and Rough Set Based Algorithms

The results of PSORS and PSOPRS using DT as the classification algorithm are presented in Table VI. The results of PSOPRSN using DT, NB and NN as the classification algorithms are shown in Table VII.

Comparing PSOCP (Table II) with PSORS and PSOPRS, it can be seen that PSOCP achieves better reduction rates, but lower classification rates than both PSORS and PSOPRS. The standard deviations of all three algorithms are comparable. However, PSORS and PSOPRS can achieve slightly higher best classification rates. Comparing PSOCPC (Table IV) with PSORS and PSOPRS, it can be observed that PSOCPC usually selected a smaller number of features and achieved similar or better classification performance than PSORS and PSOPRS.

TABLE VI
PSORS AND PSOPRS WITH DT AS THE LEARNING ALGORITHM

	Dermatology		Spect	
	#Attr	Ave (Sd, Best)	#Attr	Ave (Sd, Best)
All	33	0.828	22	0.809
PSORS	21	0.860 (0.048, 0.975)	21	0.860 (0.048, 0.975)
PSOPRS				
α = 0.9	21	0.860 (0.048, 0.975)	17.3	0.806 (0.022, 0.843)
α = 0.8	21	0.860 (0.048, 0.975)	17.5	0.800 (0.020, 0.820)
α = 0.75	21	0.860 (0.048, 0.975)	15.57	0.818 (0.008, 0.820)

	Soybean			
	#Attr	Ave (Sd, Best)		
All	35	0.819		
PSORS	21.53	0.803 (0.046, 0.872)		
PSOPRS				
α = 0.9	21.6	0.805 (0.044, 0.872)		
α = 0.8	21.67	0.805 (0.044, 0.872)		
α = 0.75	21.63	0.804 (0.043, 0.872)		

Results in Table VII show that PSOPRSN is able to produce the lowest average number of selected features, though the classification rates suffer in most cases. In comparison, the second proposed algorithm PSOCPC, according to Table IV, is not able to achieve such high feature reduction rates, but the average classification rates and standard deviations are better than PSOPRSN. Also between the two algorithms, the best classifications rates are higher in most datasets using PSOCPC.

As a result of PSOCPC considering the overall quality of a reduct as well as the quality of individual features within a reduct, it seems that it is able to strike a balance between reducing number of features and maintaining good classification rates.

TABLE VII
RESULTS OF PSOPRSN WITH α = 0.75

	Dermatology			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
γ				
0.9	8.17	0.757 (0.068, 0.918)	0.816 (0.056, 0.943)	0.787 (0.058, 0.877)
0.8	8.07	0.775 (0.078, 0.967)	0.799 (0.056, 0.959)	0.784 (0.060, 0.918)
0.75	7.73	0.743 (0.085, 0.926)	0.786 (0.064, 0.910)	0.766 (0.073, 0.893)
0.5	6.43	0.752 (0.093, 0.951)	0.783 (0.075, 0.959)	0.725 (0.083, 0.943)

	Spect			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
γ				
0.9	13.97	0.820 (0.0, 0.820)	0.767 (0.010, 0.775)	0.818 (0.010, 0.831)
0.8	8.97	0.799 (0.017, 0.820)	0.783 (0.024, 0.820)	0.834 (0.021, 0.843)
0.75	7.07	0.798 (0.012, 0.831)	0.797 (0.029, 0.843)	0.805 (0.040, 0.843)
0.5	4.63	0.786 (0.026, 0.843)	0.796 (0.025, 0.843)	0.739 (0.248, 0.843)

	Soybean			
		DT	NB	NN
	#Attr	Ave (Sd, Best)	Ave (Sd, Best)	Ave (Sd, Best)
γ				
0.9	9.7	0.714 (0.031, 0.767)	0.756 (0.036, 0.824)	0.675 (0.037, 0.749)
0.8	9	0.705 (0.038, 0.780)	0.745 (0.041, 0.846)	0.665 (0.039, 0.749)
0.75	8.77	0.713 (0.043, 0.775)	0.747 (0.031, 0.811)	0.668 (0.033, 0.749)
0.5	7.47	0.713 (0.039, 0.802)	0.761 (0.042, 0.833)	0.670 (0.033, 0.727)

TABLE VIII
RESULTS OF CFSF AND CFSB WITH DT AS THE LEARNING ALGORITHM

	Dermatology		LED Display		Spect		Soybean	
Method	Size	Accuracy	Size	Accuracy	Size	Accuracy	Size	Accuracy
CfsF	17	0.873	13	1.0	4	0.70	12	0.805
CfsB	17	0.873	13	1.0	4	0.70	14	0.854

E. Comparisons with Two Traditional Algorithms

Table VIII shows the performance of CfsF and CfsB algorithms for feature selection and DT as the learning algorithm. Due to the page limit, the results of four datasets are presented.

Compared with PSOCP, CfsF and CfsB obtain better classification rates in most datasets as well as achieve a smaller number of selected features in all datasets, but the best classification rate of PSOCP is better than CfsF and CfsB. When comparing PSOCPC with CfsF and CfsB, different α values lead to different observations. When looking at the higher α values, classification rates are similar or better in the case of Spect dataset although the number of selected features is slightly greater in PSOCPC than in both CfsF and CfsB. With the lowest α value, PSOCPC achieves better classification rates than both traditional algorithms. Furthermore, it selects a smaller subset of features in most cases.

VI. CONCLUSIONS

Two new methods using a probabilistic rough set model named DTRS and PSO are proposed for feature selection problems. The two methods, which are both based on decision-theoretic rough set, differ in the structure of their fitness functions. The first method, PSOCP, evaluates reducts based on their ability to preserve decision rules and the costs of those rules. The second method, PSOCPC, extends the first one by introducing the measure of individual confidence of features in a reduct. The methods are examined with various parameter values and compared with each other, with the three existing PSO and rough set based methods and with two traditional feature selection algorithms.

The results indicate that both methods are able to reduce the number of features while maintaining good classification rates or even improving them. PSOCPC can outperform PSOCP in terms of feature reduction rates and at the same time achieve better classification rates in half of the datasets. Compared with the three existing PSO based methods (PSORS, PSOPRS and PSOPRSN) and two traditional feature selection methods, PSOCP achieved competitive results with other methods, which is either with better classification performance or with better feature reduction rates. PSOCPC with proper parameter settings can outperform the all the other methods mentioned above. Meanwhile, the results show that PSOCP and PSOCPC as filter approaches are general to the three different classification algorithms (DT, NB and NN).

REFERENCES

[1] M. Dash and H. Liu, "Feature selection for classification," *Intelligent Data Analysis*, vol. 1, pp. 131–156, 1997.

[2] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *ARTIFICIAL INTELLIGENCE*, vol. 97, no. 1, pp. 273–324, 1997.

[3] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, 2003.

[4] A. Unler and A. Murat, "A discrete particle swarm optimization method for feature selection in binary classification problems," *European Journal of Operational Research*, vol. 206, no. 3, pp. 528–539, 2010.

[5] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *IEEE International Conference Neural Networks (ICNN'95)*, vol. 4, 1995, pp. 1942–1948.

[6] 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 Congress on Evolutionary Computation (CEC'98)*, 1998, pp. 78–83.

[7] L.-Y. Chuang, S.-W. Tsai, and C.-H. Yang, "Improved binary particle swarm optimization using catfish effect for feature selection," *Expert Syst. Appl.*, vol. 38, no. 10, pp. 12 699–12 707, 2011.

[8] C.-S. Yang, L.-Y. Chuang, C.-H. Ke, and C.-H. Yang, "Boolean binary particle swarm optimization for feature selection," in *IEEE Congress on Evolutionary Computation*, 2008, pp. 2093–2098.

[9] S. A. Rahman, Z.-A. Mohamed-Hussein, and A. A. Bakar, in *ISDA*.

[10] 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 *Australasian Conference on Artificial Intelligence*, 2012, pp. 313–325.

[11] Z. Pawlak, "Rough sets," *International Journal of Parallel Programming*, vol. 11, no. 5, pp. 341–356, 1982.

[12] X. Wang, J. Yang, X. Teng, W. Xia, and R. Jensen, "Feature selection based on rough sets and particle swarm optimization," *Pattern Recogn. Lett.*, vol. 28, no. 4, pp. 459–471, 2007.

[13] Y. Yao and Y. Zhao, "Attribute reduction in decision-theoretic rough set models," *Inf. Sci.*, vol. 178, no. 17, pp. 3356–3373, 2008.

[14] A. Frank and A. Asuncion, "Uci machine learning repository," 2010.

[15] J. Kennedy and R. Eberhart, "A discrete binary version of the particle swarm algorithm," in *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 5, 1997, pp. 4104–4108 vol. 5.

[16] Y. Yao, "Probabilistic rough set approximations," *Int. J. Approx. Reasoning*, vol. 49, no. 2, pp. 255–271, 2008.

[17] J. Yang and V. G. Honavar, "Feature subset selection using a genetic algorithm," *IEEE Intelligent Systems*, vol. 13, no. 2, pp. 44–49, 1998.

[18] B. Chakraborty, "Genetic algorithm with fuzzy fitness function for feature selection," in *IEEE International Symposium on Industrial Electronics*, vol. 1, 2002, pp. 315 – 319.

[19] A. AlSukker, R. Khushaba, and A. Al-Ani, "Enhancing the diversity of genetic algorithm for improved feature selection," in *IEEE International Conference on Systems Man and Cybernetics (SMC)*, 2010, pp. 1325 – 1331.

[20] J. Huang, Y. Cai, and X. Xu, "A hybrid genetic algorithm for feature selection wrapper based on mutual information," *Pattern Recogn. Lett.*, vol. 28, no. 13, pp. 1825–1844, 2007.

[21] I. Babaoglu, O. Findik, and E. Ulker, "A comparison of feature selection models utilizing binary particle swarm optimization and genetic algorithm in determining coronary artery disease using support vector machine," *Expert Systems with Applications*, vol. 37, no. 4, pp. 3177 – 3183, 2010.

[22] K. Neshatian and M. Zhang, "Dimensionality reduction in face detection: A genetic programming approach," in *Image and Vision Computing New Zealand, 2009. IVCNZ '09. 24th International Conference*, 2009, pp. 391 –396.

[23] ——, "Using genetic programming for context-sensitive feature scoring in classification problems," *Connect. Sci.*, vol. 23, no. 3, pp. 183–207, 2011.

[24] A. Unler and A. Murat, "A discrete particle swarm optimization method for feature selection in binary classification problems," *European Journal of Operational Research*, vol. 206, no. 3, pp. 528–539, 2010.

[25] L. Cervante, B. Xue, M. Zhang, and L. Shang, "Binary particle swarm optimisation for feature selection: A filter based approach," in *IEEE Congress on Evolutionary Computation (CEC)*, 2012, pp. 1 –8.

[26] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *IEEE International Conference on Evolutionary Computation Proceedings*, 1998, pp. 69 –73.

[27] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques (Second Edition)*. Morgan Kaufmann, 2005.

[28] M. A. Hall, "Correlation-based feature subset selection for machine learning," Ph.D. dissertation, The University of Waikato, Hamilton, New Zealand, 1999.

[29] Z. Pawlak, "Rough set theory and its applications to data analysis," *Cybernetics and Systems*, vol. 29, no. 7, pp. 661–688, 1998.