Expert Systems with Applications 39 (2012) 6123–6128

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

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

Rough set and scatter search metaheuristic based feature selection for credit scoring

Jue Wang [a,*], Abdel-Rahman Hedar [b], Shouyang Wang [a], Jian Ma [c]

[a] *Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, PR China*
[b] *Department of Computer Science, Faculty of Computers and Information, Assiut University, Egypt*
[c] *Department of Information Systems, City University of Hong Kong, Hong Kong*

ARTICLE INFO

Keywords:
Credit scoring
Feature selection
Rough set
Scatter search
Meta-heuristics

ABSTRACT

As the credit industry has been growing rapidly, credit scoring models have been widely used by the financial industry during this time to improve cash flow and credit collections. However, a large amount of redundant information and features are involved in the credit dataset, which leads to lower accuracy and higher complexity of the credit scoring model. So, effective feature selection methods are necessary for credit dataset with huge number of features. In this paper, a novel approach, called RSFS, to feature selection based on rough set and scatter search is proposed. In RSFS, conditional entropy is regarded as the heuristic to search the optimal solutions. Two credit datasets in UCI database are selected to demonstrate the competitive performance of RSFS consisted in three credit models including neural network model, J48 decision tree and Logistic regression. The experimental result shows that RSFS has a superior performance in saving the computational costs and improving classification accuracy compared with the base classification methods.

1. Introduction

Credit scoring is a method modeling potential risk of credit applications, which has experienced two decades of rapid growth with significant increases in auto-financing, credit card debt, and so on. The advantages of credit scoring include reducing the cost of credit analysis, enabling faster credit decisions, closer monitoring of existing accounts, and prioritizing collections (Brill, 1998). Traditionally, logistic regression (Henley, 1995) and discriminant analysis (Wiginton, 1980) are the most widely used approaches when assessing customer credit risk. And then a series of non-parametric methods from the machine learning, data mining, artificial intelligence and operations research communities have been employed, including *k*-nearest neighbor (Henley & Hand, 1996), genetic programming (Ong, Huang, & Tzeng, 2005), decision tree (Davis, Edelman, & Gammerman, 1992), support vector machines (Huang, Chen, Hsu, Chen, & Wu, 2004) and neural network (Malhotra & Malhotra, 2002; West, 2000).

With the growth of the credit industry and the large loan portfolios under management today, credit industry is actively developing more accurate credit scoring models. However, credit scoring datasets containing huge number of features are often involved, which leads to high complexity and be instable with high-dimensional data for many credit scoring methods. Hence, feature selection will be necessary for significantly reducing the burden of computing and improving the accuracy of the credit scoring models (Isabelle & Andre, 2003; Liu & Motoda, 1998). This effort is leading to the investigation of effective approach to feature selection for credit scoring applications.

Feature selection is one of the most fundamental problems in the field of machine learning. The main aim of feature selection is to determine a minimal feature subset from a problem domain while retaining a suitably high accuracy in representing the original features. Actually, feature selection is a process of finding a subset of features that ideally is necessary and sufficient to describe the target concept from the original set of features in a given data set.

Due to the abundance of noisy, irrelevant or misleading features, the ability to handle imprecise and inconsistent information in real world problems has become one of the most important requirements for feature selection. Rough sets theory, proposed by Pawlak, is a novel mathematic tool handling uncertainty and vagueness, and inconsistent data (Pawlak, 1982, 1991; Pawlak & Skowron, 2000). The rough set approach to feature selection is to select a subset of features (or attributes), which can predict or classify the decision concepts as well as the original feature set (Swiniarski & Andrzej, 2003). Obviously, feature selection is an attribute subset selection process, where the selected attribute subset not only retains the representational power, but also has minimal redundancy. There are many rough set algorithms for feature selection. The simplest approach is based on calculation

[*] Corresponding author.
E-mail address: wjue@amss.ac.cn (J. Wang).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.11.011

of a core for discrete attribute data set containing strongly relevant features, and reducts contain a core plus additional weakly relevant features. Mutual information and discernibility matrix based feature selection methods have been proposed in some literatures. In addition, many researchers have endeavored to develop some global optimization algorithms based on genetic algorithm, ant colony optimization, simulated annealing, Tabu search and others (Jensen & Shen, 2003, 2004; Jelonek, Krawiec, & Slowinski, 1995; Hedar, Wang, & Fukushima, 2008; Tan, 2004; Zhai et al., 2002). These techniques have been successfully applied to data reduction, text classification and texture analysis (Lin, Yao, & Zadeh, 2002).

In this paper, a novel method of feature selection based on rough sets and scatter search (RSFS) is proposed for credit scoring data. Scatter search meta-heuristic is an artificial-evolutionary-based algorithm and lies among memory-based heuristics (Glover, 1977, 1998; Glover, Laguna, & Mart, 2003; Glover & Kochenberger, 2003). However, the contributions of memory-based heuristics to information systems and data mining applications are still limited compared with other computing intelligence tools like evolutionary computing and neural networks (Osman & Kelly, 1996; Rego & Alidaee, 2005). RSFS uses a binary representation of solutions in the process of feature selection. The conditional entropy is invoked to measure the quality of solution and regarded as a heuristic. The numerical results from two credit datasets indicate that the proposed method shows a superior performance in saving the computational costs and improving the accuracy of several credit models including neural network, logistic regression and J48 decision tree.

The rest of the paper is organized as follows. In the next section, we briefly give the principles of rough set. In Section 3, we highlight the main components of RSFS and present the algorithm formally. In Section 4, numerical results of RSFS are illustrated, and the classification results are presented comparing with the models without feature selection. Finally, the conclusion makes up Section 5.

2. Rough sets preliminaries

An information system is a formal representation of a dataset to be analyzed, and it is defined as a pair $S = (U, \mathbb{A})$, where U is a non-empty set of finite objects, called the universe of discourse, and $\mathbb{A}$ is a non-empty set of attributes. With every attribute $a \in \mathbb{A}$, a set of its values V_a is associated (Pawlak, 1991). In practice, we are mostly interested in dealing with a special case of information system called a decision system. It consists of a pair $S = (U, \mathbb{C} \cup \mathbb{D})$, where $\mathbb{C}$ is called a conditional attributes set and $\mathbb{D}$ a decision attributes set.

The RS theory is based on the observation that objects may be indiscernible (indistinguishable) because of limited available information. For a subset of attributes $P \subseteq \mathbb{A}$, the indiscernibility relation is defined by $IND(P)$ (Pawlak, 1991):

$$IND(P) = \{(\xi, \eta) \in U \times U | \forall a \in P, a(\xi) = a(\eta)\}.$$

It is easily shown that $IND(P)$ is an equivalence relation on the set U. The relation $IND(P)$, $P \subseteq \mathbb{A}$, constitutes a partition of U, which is denoted $U/IND(P)$. If $(\xi, \eta) \in IND(P)$, then ξ and η are indiscernible by attributes from P. The equivalence classes of the P-indiscernibility relation are denoted $[\xi]_P$. For a subset $\Xi \subseteq U$, the P-lower approximation and P-upper approximation of Ξ can be defined as follows, respectively,

$$\underline{P}\Xi = \{\xi | [\xi]_P \subseteq \Xi\},$$
$$\overline{P}\Xi = \{\xi | [\xi]_P \cap \Xi \neq \emptyset\}.$$

As an illustrative example, Table 1(i) shows a dataset which consists of three conditional attributes $\mathbb{C} = \{a, b, c\}$, one decision attribute $\mathbb{D} = \{d\}$, and six objects $U = \{e1, e2, \ldots, e6\}$. If $P = \{a,b\}$, then objects $e1$, $e2$, and $e3$ are indiscernible, and so are objects $e4$ and $e6$. Thus, $IND(P)$ yields the following partition of U:

$$U/IND(P) = \{\{e1, e2, e3\}, \{e4, e6\}, \{e5\}\}.$$

Table 1
An example of reducts.

(i) A dataset					(ii) A reduced dataset				(iii) A reduced dataset			
U	*a*	*b*	*c*	*d*	*U*	*a*	*c*	*d*	*U*	*b*	*c*	*d*
*e*1	0	0	0	1	*e*1	0	0	1	*e*1	0	0	1
*e*2	0	0	1	0	*e*2	0	1	0	*e*2	0	1	0
*e*3	0	0	2	0	*e*3	0	2	0	*e*3	0	2	0
*e*4	1	0	0	1	*e*4	1	0	1	*e*4	0	0	1
*e*5	1	1	1	1	*e*5	1	1	1	*e*5	1	1	1
*e*6	1	0	2	0	*e*6	1	2	0	*e*6	0	2	0

For example, if $\Xi = \{e1, e4, e5\}$, then $\underline{P}\Xi = \{e5\}$, $\overline{P}\Xi = \{e1, e2, e3, e4, e5, e6\}$; if $\Xi = \{e2, e3, e6\}$, then $\underline{P}\Xi = \phi$, $\overline{P}\Xi = \{e1, e2, e3, e4, e6\}$.

For an information system $S = (U, \mathbb{A})$ and a partition of U with classes X_i, $1 \leqslant i \leqslant n$, the entropy of attribute set $B \subseteq A$ is defined as (Pal, Uma Shankar, & Mitra, 2005)

$$H(B) = -\sum_{i=1}^{n}(p(X_i))log(p(X_i))$$

where $IND(B) = \{X_1, X_2, \ldots, X_n\}$, $p(X_i) = |X_i|/|U|$, and $|\cdot|$ is the cardinality.

The conditional entropy of an attribute set D with reference to another attribute set B is defined as follows:

$$H(D|B) = -\sum_{i=1}^{n}(p(x_i))\sum_{j=1}^{m}(p(Y_j|X_i))log(p(Y_j|X_i))$$

where $p(Y_j|X_i) = |Y_j \cap X_i|/|X_i|$, and $U/IND(D) = \{Y_1, Y_2, \ldots, Y_m\}$, $U/IND(B) = \{X_1, X_2, \ldots, X_n\}$, $1 \leqslant i \leqslant n$, $1 \leqslant j \leqslant m$. And some theorems have been proved in some literatures.

Theorem 2.1. *$H(D|B) = H(D \cup B)) - H(B)$.*

Theorem 2.2. *If P and Q are the attribute sets of an information system S = (U,A) and IND(Q) = IND(P), then H(Q) = H(P).*

Theorem 2.3. *If P and Q are the attribute sets of an information system S = (U,A), $P \subseteq Q$ and H(Q) = H(P), then IND(Q) = IND(P).*

Theorem 2.4. *An attribute r in an attribute set $R \subseteq C$ is reducible if and only if $H(r|R - \{r\}0 = 0$.*

Theorem 2.5. *Given a relatively consistent decision table $S = (U, C \cup D)$, an attribute set R is relatively independent if and only if $H(D|R) = H(D|R - r)$ for every $r \in R$.*

Theorem 2.6. *Given a relatively consistent decision table $S = (U, C \cup D)$, an attribute set $R \subseteq C \subseteq A$ of an information system S = (U,A) is a reduct of B if and only if*

(1) *H(R) = H(C).*
(2) *The attribute set R is relatively independent.*

Consider the dataset shown in Table 1(i), and let $P = \{a,b\}$ and $Q = \{d\}$. Then

$$U/IND(Q) = \{\{e1, e4, e5\}, \{e2, e3, e6\}\},$$
$$U/IND(P) = \{\{e1, e2, e3\}, \{e4, e6\}, \{e5\}\},$$
$$H(Q|P) = 0.7925.$$

One of the major applications of rough set theory is the attribute reduction, that is, the elimination of attributes considered to be redundant, while avoiding information loss (Pawlak, 1991; Pawlak & Skowron, 2000). The reduction of attributes is achieved by comparing equivalence relations generated by sets of attributes. Using the conditional entropy as a measure, attributes are removed so that the reduced set provides the same entropy value as the original. In a decision system, a reduct is formally defined as a subset R of the conditional attribute set $\mathbb{C}$ such that $H(R) = H(C)$, where $\mathbb{D}$ is the decision attributes set. A given dataset may have many reducts. Thus the set $\mathfrak{R}$ of all reducts is defined as (Pawlak, 1991):

$$\mathfrak{R} = \{R : R \subseteq \mathbb{C}; H(R) = H(C)\}.$$

The intersection of all the sets in $\mathfrak{R}$ is called the core,

$$\mathrm{Core}(\mathfrak{R}) = \bigcap_{R \in \mathfrak{R}} R.$$

The elements of the core are those attributes that cannot be eliminated without introducing more contradictions to the dataset. In the process of attribute reduction, a set $\mathfrak{R}_{min} \subseteq \mathfrak{R}$ of reducts with minimum cardinality is searched for:

$$\mathfrak{R}_{\min} = \{R \in \mathfrak{R} : |R| \leqslant |S|,\ \forall S \in \mathfrak{R}\}.$$

which is called the minimal reduct set.

Using the example shown in Table 1(i), the entropy of $\mathbb{D} = \{d\}$ on all possible subsets of $\mathbb{C}$ can be calculated as:

$$\begin{array}{lll} & H(D|C) = 0, & \\ H(D|\{a,b\}) = 0.7925, & H(D|\{a,c\}) = 0, & H(D|\{b,c\}) = 0, \\ H(D|\{a\}) = 0.9183, & H(D|\{b\}) = 0.8091, & H(D|\{c\}) = 0.3333. \end{array}$$

So, the minimal reduct set for this example is given by:

$$\mathfrak{R}_{\min} = \{\{a,c\},\{b,c\}\}.$$

If the minimal reduct $\{a,c\}$ is chosen, then the example dataset shown in Tabel 1(i) can be reduced as in Table 1(ii). On the other hand, Table 1(iii) shows a reduced dataset corresponding to the minimal reduct $\{b,c\}$.

It is well known that finding a minimal reduct is NP-hard (Pawlak, 1991). The most primitive solution to locating such a reduct is to simply generate all possible reducts and choose some with minimal cardinality. Obviously, this is an expensive procedure and it is only practical for simple datasets. For most of the applications, only one minimal reduct is required, so all the calculations involved in discovering the rest are pointless. Therefore, an alternative strategy is required for large datasets.

3. Credit scoring based on rough sets and scatter search

In this section, the proposed feature selection method for credit scoring is presented, which is based on rough set and scatter search. Three credit scoring models and several performance measures are involved in the illustration of the proposed method.

3.1. Feature selection based on rough set and scatter search

Scatter search (SS) is an evolutionary algorithm or population-based algorithm, which was first proposed by F. Glover in the 1970s (Glover, 1977). Actually, it is based on some of the results dating back to the 1960s. However, the SS template in its final form was given in the literature (Glover, 1998). SS, unlike most of other evolutionary algorithms, stores solutions in a so called "*reference set*" (*RefSet*)[1] and constructs new solutions by combining existing ones.

Moreover, SS has more flexible framework than the other evolutionary algorithms and uses a memory-type diversification procedure for more efficient globally search (Laguna & Mart, 2003). The scatter search meta heuristic, because of its capability in solving both combinatorial and continuous optimization problems, has successfully been applied to a widespread variety of hard optimization problems such as assignment problems, binary problems and non-linear optimization. In the original proposal, Glover described scatter search as a method that uses a succession of coordinated initializations to generate solutions. He introduced the reference set (*RefSet*) of solutions and several guidelines, including that the search takes place in a systematic way as oppose to the random designs of other methods. In the following section, structure of RSFS is presented in details.

RSFS uses a binary representation for solutions (feature subsets). Therefore, a trial solution x is a 0-1 vector with dimension equal to the number of conditional attributes $|\mathbb{C}|$. If a component x_i of x, $i = 1, \ldots, |\mathbb{C}|$, has the value 1, then the ith attribute is contained in the attribute subset represented by the trial solution x. Otherwise, the solution x does not contain the ith attribute.

The entropy $H(\mathbb{D}|x)$ of decision attribute $\mathbb{D}$ is used to measure the quality of a solution x. Comparing two solution x and x', we say x is better than x' if one of the following conditions holds:

- $H(\mathbb{D}|x) < H(\mathbb{D}|x')$.
- $|x_i| < |x'_i|$ if $H(\mathbb{D}|x) = H(\mathbb{D}|x')$.

where $H(\mathbb{D}|x)$ is the entropy of $\mathbb{D}$ reference to x, and $|x|$ is the cadinality of x.

RSFS starts with "diversification generation procedure" to generate population P which follows Glover's systematic procedure for generating diverse 0-1 vectors. Then the "solution improvement procedure" in the following is recalled to refine the selected solutions in P. Then, the initial *RefSet* is constructed from the improved population P. In the RSFS process of finding the optimal feature subset, any subset which has been visited is saved in a set called *Reduct Set* (*RedSet*). A vector of dimension $|\mathbb{C}|$ counts the numbers of appearing of each conditional attribute in *RedSet*. Finally, an intensification procedure is applied to refined the best obtained solutions. Specifically, *Best Reduct Shaking* and *Elite Reducts Inspiration* mechanisms are used for this job as follows. The framework of RSFS is shown as follows and the detail algorithm of RSFS can be found in Wang et al. (2009).

Algorithm: feature selection method based on rough set and scatter search

```
Begin
Diversification Generation
Solution Improvement
   while (Stopping criterion not met)
do if (NewSolutions = TRUE)
then
 GenerateSubsets
CombineSolutions
else Generate Diversified Solutions
endif
Improve Solutions
Update Reference Set
end
end
Best Reduct Shaking
Elite Reducts Inspiration
```

[1] *RefSet* corresponds to the term *population* in other EAs.

Diversification Generation: Let Population *P* be a set of diverse trial solutions. Frequency-based memory is employed to generate diverse solutions in this strategy.

Solution Improvement: Let V^F be a vector counting the number of appearing of each conditional attribute in *Redset*. Set NewSolution $x' := x$, if x is a reduct,remove the attribute form x' with the minimum frequency in V^F; otherwise, add to x' the attribute that has the maximum frequency in V^F.

GenerateSubsets: generates all pairs of solutions (x,y) in *RefSet*. It is noteworthy that the "subset generation procedure" discards all those pairs of reference solutions which have already been combined in previous iterations.

CombineSolutions: For each subset $\{x,y\}$, one child solution *z* is generated as follows:

$$z_i = \begin{cases} 1, & \text{if } \zeta_i \geqslant r; \\ 0, & \text{if } \zeta_i < r, \end{cases}$$

where *r* is a random number in the interval (0,1) and $\zeta_i = \frac{H(\mathbb{D}|x_i)+H(\mathbb{D}|y_i)}{H(\mathbb{D}|x)+H(\mathbb{D}|y)}$, $i = 1, \ldots, |\mathbb{C}|$.

Reference set update: Update *RefSet* to have the best μ_1 solutions from the old *RefSet* and the improved generated children, and μ_2 diverse solutions chosen randomly from *P*, where $\mu_1 + \mu_2 = \mu$.

Best Reduct Shaking. SSAR tries to reduce the attributes contained in the best obtained reduct x^{best} one by one without increasing $H(\mathbb{D}|x^{best})$.

Elite Reducts Inspiration. A trial solution x^{ERI} is constructed as the intersection of the n_R best reducts in *RedSet*, where n_R is a prespecified number. If the number of attributes involved in x^{ERI} is less than that in x^{best} by at least two, then the zero position in x^{ERI} which gives the lowest *H*-value is updated to be one. This mechanism is continued until the number of attributes involved in x^{ERI} becomes less than that in x^{best} by one.

3.2. Credit scoring based on RSRF

This section investigates the credit scoring accuracy of three credit scoring models without feature selection and with feature selection based on rough set and scatter search, including radial basis function (RBF), logistic regression and J48 decision tree. They use different algorithms to estimate the unknown credit scoring function and employ different training methods to acquire information from the credit scoring examples available.

Artificial neural networks has been widely used in credit scoring problems, and it has been reported that its accuracy is superior to the traditional statistical methods. RBF is the most frequently used neural network architecture in commercial applications including credit scoring. Logistic regression model is one of the most popular statistical tools for classification problems. Logistic regression model, unlike other statistical tools (e.g. discriminant analysis or ordinary linear regression), can suit various kinds of distribution functions such as Gamble, Poisson, normal, etc. and is more suitable for the credit scoring problems. J48 is also applied to credit scoring, which is a predictive machine-learning model. It decides the target value (dependent variable) of a new sample based on various attribute values of the available data. The internal nodes of a decision tree denote the different attributes, the branches between the nodes tell us the possible values that these attributes can have in the observed samples, while the terminal nodes tell us the final value (classification) of the dependent variable.

Depending on the credit features selected and the parameters setting of models, the estimation of the credit scoring function may result in different classification results. The credit scoring models are tested using 10-fold cross validation with two real world data set, Australian credit data and Japanese credit data. Each of the *k* subsets acted as an independent holdout test for the model trained with the remaining $k - 1$ subsets. The advantage of cross validation are that all of the test sets were independent and the reliability of the results could be improved. A typical experiment uses $k = 10$, then we divide each credit data into 10 subsets for cross validation and carried out our implementation on 10 training data sets.

3.3. Performance measure

Given a classifier and an instance, there are many possible outcomes. If the instance is positive and it is classified as positive, it is counted as a true positive (TP); if the instance is negative and it is classified as positive, it is counted as a false positive (FP). A TP rate is the percentage of correct system responses during those periods when the user intends control and a FP rate is the percentage of incorrect system responses during the No Control periods. These criteria are positively correlated and our aim is to maximize the TP for a reasonably low fixed FP rate. The TP rate (true positive rate), FP rate (false positive rate), Precision, recall and *F*-measure are expressed as:

$$tprate = \frac{TP}{P}; \quad fprate = \frac{FP}{N}; \quad \text{Precision} = \frac{TP}{TP+FP}; \quad \text{Recall} = \frac{TP}{P}$$

$$F\text{-measure} = \frac{2}{1/\text{precision} + 1/\text{recall}}$$

where *P* is the total positives and *N* is the total negatives.

In addition to above performance measure, ROC curve is usually used to measure the performance of the models (Zhou, Lai, & Yen, 2009). The ROC graph is a useful technique for ranking models and visualizing their performance. The ROC is a two-dimensional graph in which true positive rate is plotted on the *Y*-axis and false positive rate is plotted on the *X*-axis. AUC is the area under the ROC curve. Generally, a model with a larger AUC will have a better average performance. One point in ROC space is better than another if it is to the northwest.

4. Numerical experiments

The proposed approach has been applied for feature selection and credit scoring on two benchmark data sets from real world. The methodology we followed is consisted in two cases: (1) feature selection in order to obtain the best reduct for credit scoring and (2) measuring the accuracy of three credit classifier. This research investigates the potential of RSFS for credit risk management.

4.1. Data set description and parameters setting

Two real world credit data sets derived from various applications are used to evaluate the performance of the proposed algorithm. These data sets are made public from the UCI Repository of Machine Learning Databases, and are mostly used to compare the performance of various classification models. In order to facilitate the following discussion, a brief description of these data sets is listed in Table 2.

The Australian credit dataset contains 690 instances, of which 307 correspond to creditworthy applicants and 383 correspond to applicants to whom credit should be refused. Each instance is described by 14 attributes. Six attributes are continuous while the remaining are categorical. In order to preserve the confidentiality of the data, the names and values of the attributes were replaced by meaningless identifiers.

The second experimental dataset in this subsection is about Japanese consumer credit card application approval. It contains 664 instances, which is described by 15 attributes. For confidentiality

Table 2
Description of two credit datasets in UCI.

Country	Attributes	Sample size	Good credit	Bad credit (%)
Australian	14	690	307	383
Japanese	15	664	299	365

all attribute names and values have been changed to meaningless symbols.

For these two credit datasets, the RSFS was coded in MATLAB and all tests were performed on an Intel Celeron 2.2 GHz processor under the Microsoft Windows XP operating system. The RSFS code was run 30 times with different initial solutions. Before discussing these results, we summarize all parameters used in the RSFS algorithm with their assigned values in Table 3. The performance of RSFS was tested using different values of these parameters. First, based on the common setting in the literature, the population size $|P|$ was set to $2|\mathbb{C}|$. The size r of *RefSet* is usually small compared with the other EA evolutionary algorithms and it is usually no more than 20. So we set the size of *Refset* was 10 in this paper. Especially the values for u_1 and u_2 equal to 8 and 2, respectively, proved to provide good results. According to our preliminary experiments, the number of the best reducts used to compute x^{ERI} was 3 and the maximal number of generations with 50 is enough to obtain stable result.

4.2. Feature selection and credit scoring for two credit datasets

This section provides a discussion of the experimental procedure and the results of the experiments.

In Table 4, the result of feature selection based on RSFS for Australian and Japanese credit datasets are illustrated, which will be contribute to decrease the complexity of classification and improve the accuracy of credit scoring model.

Three strategies were used in this study, namely RBF, J48 and Logistic. A comparison was made between using all original features and using the features pre-selected by RSFS. The results for the two data sets were obtained and summarized in Tables 5 and 6 respectively.

For the German credit data, it is evident from Table 5 that logistic regression has the best performance in terms of the listed evaluation criteria including TP Rate, FP rate, Precision, Recall and *F*-measure, closely followed by neural network model and J48 decision tree. Similarly, among the listed models with RSFS, logistic regression with RSFS still outperforms other two models, whose classification accuracy can achieve as high as 88.9%.

As also can be seen from this table, there is significant improvement in performance between classifiers with RSFS and the base classifier. For Australian dataset, the classifiers with RSFS can produce higher TP rate, Precision, Recall and *F*-measure and lower FP rate. Simultaneously, there is a marginal decreasing in computational time when two classifiers are compared. This shows that RSFS is very promising for handling credit risk analysis.

The results for Japanese credit dataset exhibit some of the same patterns previously discussed for the Australian credit data. Logistic regression has the highest classification accuracy with 90.5%, followed closely by J48 decision tree (87.4%) and neural network model (77.9%). After feature selection using RSFS, there are only seven attributes for computing the credit score, so the cost is decreasing sharply. The saved computing time is 44.0%, 23.5% and 16.7% of original time, respectively. Although less features are involved, the precision of these three models after feature selection can achieved higher classification accuracy with 90.5%, 83.4% and 87.9%, respectively. The models with fewer attributes should not only save computing cost, but also slightly improve the accuracy for Japanese credit dataset.

In order to make further performance comparison of all the models, we use the area under the receiver operating characteristic ROC curve as another performance measurement. Taking Australian dataset as an example, Figs. 1–3 showed the performance of all models in the ROC space. Specially, the ROC area of base Models is the area of the shaded part, which is smaller than the area of the

Table 3
SSAR parameter setting.

Parameter	Definition	Value
$\|P\|$	Population size	$2\|\mathbb{C}\|$
μ	Size of *RefSet*	10
μ_1, μ_2	Sizes used to update *RefSet*	8, 2
n_R	Number of the best reducts used to compute x^{ERI}	3
I_{max}	Max number of generations	50

Table 4
Result of feature selection.

Dataset	Overall attribute	RSFS result
Australian	14	A4, A5, A7, A8, A10, A13, A14
Japanese	15	A3, A4, A6, A8, A9, A11, A15

Table 5
Performance comparison for different models in Australian dataset.

Model	RSFS	TP rate	FP rate	Precision	Recall	*F*-measure	Time-saving (%)
Logistic	NO	0.879	0.113	0.885	0.879	0.879	46.7
	YES	0.889	0.105	0.893	0.889	0.889	
RBF	NO	0.860	0.151	0.862	0.860	0.859	22.7
	YES	0.874	0.124	0.875	0.874	0.875	
J48	NO	0.855	0.140	0.858	0.855	0.855	57.1
	YES	0.884	0.116	0.884	0.884	0.884	

Table 6
Performance comparison for different models in Japanese dataset.

Model	RSFS	TP rate	FP rate	Precision	Recall	*F*-measure	Time-saving (%)
Logistic	NO	0.899	0.096	0.902	0.899	0.900	44.0
	YES	0.905	0.088	0.909	0.905	0.905	
RBF	NO	0.779	0.256	0.797	0.779	0.771	23.5
	YES	0.834	0.181	0.836	0.834	0.833	
J48	NO	0.874	0.134	0.875	0.874	0.874	16.7
	YES	0.879	0.119	0.881	0.879	0.880	

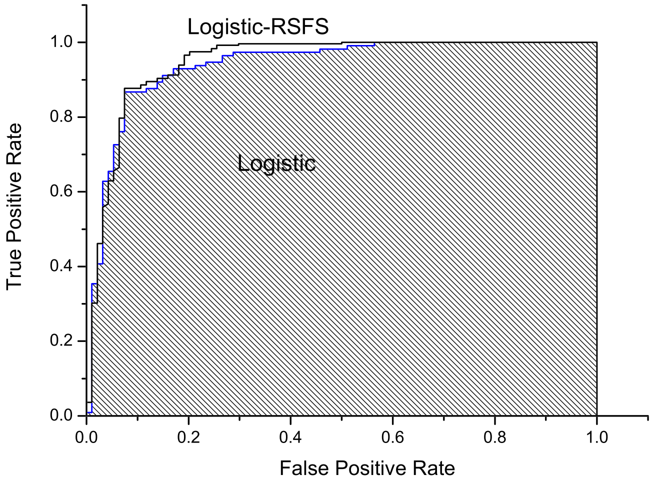

Fig. 1. ROC curve of logistic models for Australian dataset.

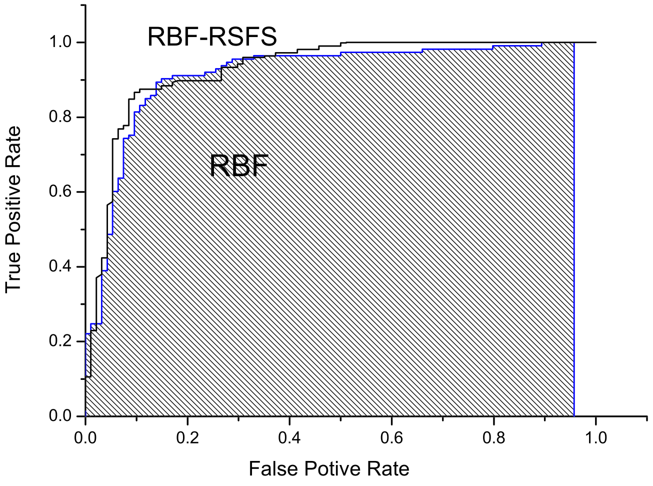

Fig. 2. ROC curve of RBF models for Australian dataset.

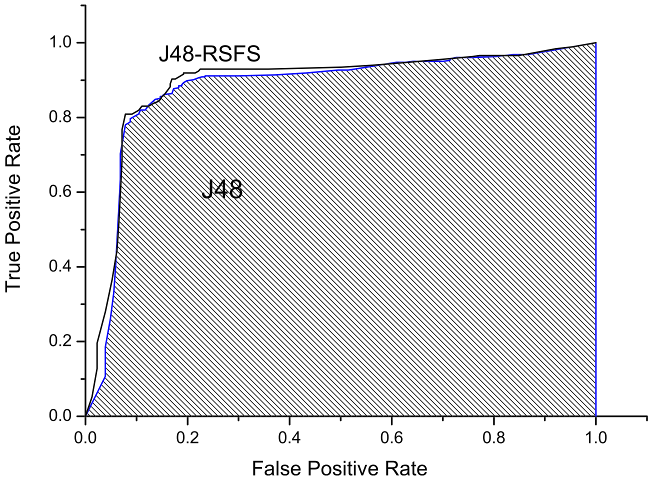

Fig. 3. ROC curve of J48 models for Australian dataset.

models with RSFS. It is illustrated that all the three classification models with RSFS has a better performance than be base models for Australian datasets.

5. Conclusion

A novel approach to feature selection based on rough set and scatter search has been proposed for credit scoring problem. New diversification and intensification elements have been embedded in RSFS to achieve better performance and to fit the considered problem. Numerical experiment on two real word credit dataset has been presented to illustrate the efficiency of RSFS. Comparisons with base credit models have revealed that RSFS is promising and it is less expensive in computational cost. This research is leading to the effort for developing more refined and accurate credit scoring model.

Acknowledgements

This work is supported by Grants from the National Natural Science Foundation of China (NSFC No. 70801058), and Strategic Research Grant of City University of Hong Kong.

References

Brill, J. (1998). The importance of credit scoring models in improving cash flow and collections. *Business Credit* (1), 16–27.

Davis, R. H., Edelman, D. B., & Gammerman, A. J. (1992). Machine learning algorithms for credit-card applications. *Journal of Mathematics Applied in Business and Industry, 4*, 43C51.

Glover, F. (1977). Heuristics for integer programming using surrogate constraints. *Decision Sciences, 8*(1), 156–166.

Glover, F. (1998). A template for scatter search and path relinking. In J.-K. Hao, E. Lutton, E. Ronald, M. Schoenauer, & D. Snyers (Eds.), *Artificial evolution. Lecture notes in computer science* (1363, pp. 125–137). Springer.

Glover, F., & Kochenberger, G. A. (Eds.). (2003). *Handbook of metaheuristics*. Boston: Kluwer Academic Publishers.

Glover, F., Laguna, M., & Mart, R. (2003). Scatter search and path relinking: Advances and applications. In F. Glover & G. A. Kochenberger (Eds.), *Handbook of metaheuristics* (pp. 1–5). Boston: Kluwer Academic Publishers.

Hedar, A., Wang, J., & Fukushima, M. (2008). Tabu search for attribute reduction in rough set theory. *Soft Computing, 12*, 909–918.

Henley, W. E. (1995). *Statistical aspects of credit scoring*. Dissertation, The Open University Milton Keynes, UK.

Henley, W. E., & Hand, D. J. (1996). A *k*-nearest neighbor classifier for assessing consumer credit risk. *Statistician, 44*(1), C77–C95.

Huang, Z., Chen, H., Hsu, C.-J., Chen, W.-H., & Wu, S. (2004). Credit rating analysis with support vector machines and neural networks: A market comparative study. *Decision Support Systems, 37*(4), 543C558.

Isabelle, G., & Andre, E. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research, 3*(7), C1157–C1182.

Jelonek, J., Krawiec, K., & Slowinski, R. (1995). Rough set reduction of attributes and their domains for neural networks. *Computational Intelligence, 11*, 339–347.

Jensen, R., & Shen, Q. (2003). Finding rough set reducts with ant colony optimization. In *Proceedings of the 2003 UK workshop on computational intelligence* (pp. 15–22).

Jensen, R., & Shen, Q. (2004). Semantics-preserving dimensionality reduction: rough and fuzzy-rough-based approaches. *IEEE Transactions on Knowledge and Data Engineering, 16*(12), 1457–1471.

Laguna, M., & Mart, R. (2003). *Scatter search methodology and implementations in C*. Boston: Kluwer Academic Publishers.

Lin, T. Y., Yao, Y. Y., & Zadeh, L. A. (2002). *Data mining, rough sets and granular computing*. Berlin: Springer-Verlag.

Liu, H., & Motoda, H. (1998). *Feature selection for knowledge discovery and data mining*. Kluwer Academic Publishers.

Malhotra, R., & Malhotra, D. K. (2002). Differentiating between good credits and bad credits using neuro-fuzzy systems. *European Journal of Operational Research, 136*(1), 190C211.

Ong, C.-S., Huang, J.-J., & Tzeng, G.-H. (2005). Building credit scoring models using genetic programming. *Expert Systems with Applications, 29*(1), C41–C47.

Osman, I. H., & Kelly, J. P. (1996). Metaheuristics: An overview. In I. H. Osman & J. P. Kelly (Eds.). *Meta-heuristics: Theory and applications* (pp. C1–C21). Boston: Kluwer Academic Publishers.

Pal, S. K., Uma Shankar, B., & Mitra, P. (2005). Granular computing, rough entropy and object extraction. *Pattern Recognition Letters, 26*(16), 2509–2517.

Pawlak, Z. (1982). Rough sets. *International Journal of Computer and Information Sciences, 11*, 341–356.

Pawlak, Z. (1991). *Rough sets: Theoretical aspects of reasoning about data*. Dordrecht, The Netherlands: Kluwer Academic Publishers.

Pawlak, Z., & Skowron, A. (2000). Rough set methods and applications: New developments in knowledge discovery in information systems. In L. Polkowski, T. Y. Lin, & S. Tsumoto (Eds.). *Studies in fuzziness and soft computing* (vol. 56). Berlin: Physica-Verlag.

Rego, C., & Alidaee, B. (2005). *Metaheursitic optimization via memory and evolution*. Berlin: Springer-Verlag.

Swiniarski, R. W., & Andrzej, S. (2003). Rough set methods in feature selection and recognition. *Pattern Recognition Letters, 24*(6), C833–C849.

Tan, S. (2004). A global search algorithm for attributes reduction. In Webb, G. I., & Yu, X. (Eds.), *AI 2004: Advances in artificial intelligence, LNAI* (Vol. 3339, pp. 1004–1010).

Wang, J., Hedar, A., Zheng, G. H., & Wang, S. Y. (2009). Scatter search for rough set attribute reduction. In *International joint conference on computational sciences and optimization, Sanya, China*.

West, D. (2000). Neural network credit scoring models. *Computers and Operations Research, 27*(11C12), 1131C1152.

Wiginton, J. C. (1980). A note on the comparison of logit and discriminant models of consumer credit behaviour. *Journal of Financial Quantitative Analysis, 15*, C757–C770.

Zhai, L. Y. et al. (2002). Feature extraction using rough set theory and genetic algorithms: An application for the simplification of product quality evaluation. *Computers & Industrial Engineering, 43*, 661–676.

Zhou, L. G., Lai, K. K., & Yen, J. (2009). Credit scoring models with AUC maximization based on weighted SVM. *International Journal of Information Technology and Decision Making, 8*(4), 677–696.