Operations Research Perspectives 3 (2016) 32–42

Contents lists available at ScienceDirect

Operations Research Perspectives

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

Application of a rule extraction algorithm family based on the Re-RX algorithm to financial credit risk assessment from a Pareto optimal perspective

Yoichi Hayashi*

Dept. of Computer Science, Meiji University Tama-ku, Kawasaki, Kanagawa 214-8571, Japan

ARTICLE INFO

Keywords:
Credit risk assessment
Credit scoring
Rule extraction
Pareto optimal
Re-RX algorithm
Financial application

ABSTRACT

Historically, the assessment of credit risk has proved to be both highly important and extremely difficult. Currently, financial institutions rely on the use of computer-generated credit scores for risk assessment. However, automated risk evaluations are currently imperfect, and the loss of vast amounts of capital could be prevented by improving the performance of computerized credit assessments. A number of approaches have been developed for the computation of credit scores over the last several decades, but these methods have been considered too complex without good interpretability and have therefore not been widely adopted. Therefore, in this study, we provide the first comprehensive comparison of results regarding the assessment of credit risk obtained using 10 runs of 10-fold cross validation of the Re-RX algorithm family, including the Re-RX algorithm, the Re-RX algorithm with both discrete and continuous attributes (Continuous Re-RX), the Re-RX algorithm with J48graft, the Re-RX algorithm with a trained neural network (Sampling Re-RX), NeuroLinear, NeuroLinear+GRG, and three unique rule extraction techniques involving support vector machines and Minerva from four real-life, two-class mixed credit-risk datasets. We also discuss the roles of various newly-extended types of the Re-RX algorithm and high performance classifiers from a Pareto optimal perspective. Our findings suggest that Continuous Re-RX, Re-RX with J48graft, and Sampling Re-RX comprise a powerful management tool that allows the creation of advanced, accurate, concise and interpretable decision support systems for credit risk evaluation. In addition, from a Pareto optimal perspective, the Re-RX algorithm family has superior features in relation to the comprehensibility of extracted rules and the potential for credit scoring with Big Data.

1. Introduction

1.1. Background

Within the field of financial analysis, the assessment of credit risk has historically been both of the utmost importance and quite difficult. Beginning in the late twentieth century, the advent of sophisticated electronic data storage technologies meant that financial institutions could readily store information regarding potential customers, such as repayment characteristics. As a result, the process of loaning capital within the United States, the United Kingdom, and other industrialized nations is largely predicated on the use of computer-generated credit scores [1].

This automated calculation of credit scores is markedly superior in a number of respects compared with the former process of hand-calculated risk assessments by banking professionals. Advantages include increased objectivity and reliability, as well as reduced costs and labor during the assessment of new credit applications [2]. Nonetheless, at present, the credit evaluation performance of humans with sufficient expertise can still be superior to the results of automated assessments. For this reason, and because the finance industry relies on the appropriate prediction of lending risks, research aimed at improving the validity of computerized credit assessments is ongoing [3].

As noted, automated lending risk evaluations are currently imperfect, and, in fact, the failure of credit scoring algorithms to identify loan recipients who will eventually default on their loans results in sizable losses [1]. Based on readily available data, Finlay determined that, as of the end of 2014, consumer debt in the United States and the United Kingdom stood at $3.1 trillion [4] and

* Corresponding author. Fax: (+81) 44-931-5161.
E-mail address: hayashiy@cs.meiji.ac.jp

http://dx.doi.org/10.1016/j.orp.2016.08.001

£297 billion [5], respectively. In addition, these two nations had annualized credit card and personal loan write-off rates of 3.03% [6] and 2.3% [7], respectively.

These data show that the loss of vast amounts of capital could be prevented by improving, even to a small extent, the performance of computerized credit assessments. As a result, numerous computational methods, including linear discriminant analysis (LDA), logistic regression (LR), and multiple discriminant analysis (MDA), have been considered for application in automated loan decision-making processes.

1.2. Credit scoring algorithm

A number of alternative approaches have been developed for the computation of credit scores over the last several decades, including support vector machines (SVMs) [8], neural networks (NNs) [3,9], ensemble classifiers [10] and various genetic algorithms [11]. However, these methods have not been widely adopted because they are considered to be more complex and to require greater resources while offering less interpretability, even though they have been shown to produce significantly better results compared with those obtained from LDA, LR, and MDA.

When considering consumer credit, it is important to note that loan decisions must be defensible for social and legal reasons. For example, according to the United States Equal Credit Opportunity Act (1976), financial institutions must be able to adequately explain why a loan application is denied [8].

1.3. NN ensembles

Pronounced improvements in the generalization capabilities of artificial NN (ANN)-based learning systems have been demonstrated via the application of ANN ensembles, based on combining the predictions of numerous trained ANNs in a voting process [12].

More recently, compared with other methods of prediction, improved generalization has been exhibited by back-propagation NN (BPNN) ensembles [13]; however, due to their lack of transparency, a factor that has greatly restricted the applications of this method, it is not possible to obtain an intuitive understanding of the decision-making processes of such ensembles [14].

1.4. Rule extraction

One line of research that shows considerable promise is rule extraction, whereby a set of simple and comprehensible rules are found to explain the behavior of NNs [9] and SVMs [8,15].

Along these lines, the following three rule extraction methods by an NN for the purpose of evaluating credit risk were assessed by Baesens et al. [9]: the NeuroRule [16], Trepan [17], and Nefclass [18] algorithms. In addition, a number of methods have been proposed for the extraction of information from BPNNs [19–21].

Since the practical application of classification processes typically has input composed of both discrete and continuous data, or so-called mixed data [22], rule extraction by Neurorule and similar programs requires that the continuous attributes first be discretized. During discretization, the input space is divided into hyper-rectangular regions, each of which corresponds to data samples belonging to a specific class, and is associated with an extracted rule condition [22].

However, some NN rule extraction programs, such as GLARE [23] and OSRE [24], do not need to discretize the attributes of continuous input data [19,20]. In such cases, linear combinations of the appropriate input attributes, incorporating both continuous and discrete attributes, are used to generate extracted rules.

Rule generation by NN ensembles is also the basis of the Discretized Interpretable Multi-Layer Perceptron (DIMLP) model [25], in which the knowledge contained in activation neurons and connections is elucidated through the use of symbolic rules. Similarly, the Rule Extraction from Network Ensemble (REFNE) [26] technique has been proposed as a means of obtaining symbolic rules from NN ensembles trained to carry out classifications. This approach extracts rules from instances generated using trained ensembles.

1.5. Rule extraction for credit risk assessment

An increasing number of bank collapses, accompanied by massive losses in the financial sector, has resulted in stricter international banking regulations and created a demand for more accurate models for assessing credit risk and structuring loan portfolios among financial institutions.

A useful credit scoring model achieves a good balance between accuracy and comprehensibility. In this context, the former refers to strong classification performance that minimizes prediction error, while the latter refers to ease of comprehension by the users. Historically, accuracy has been the primary focus of credit scoring because any improvement—regardless of how minor—can potentially lead to considerable savings and profits in the future. Therefore, a considerable amount of literature has focused on evaluating techniques to increase the accuracy of credit scoring models.

However, it is also vital that credit scoring models be comprehensible for the following reasons. First, managers need credit scoring models that are easy to interpret to justify their reasons for accepting or denying credit, which is an industry requirement in numerous countries. Second, such models reduce the reluctance among managers to use statistical techniques for credit evaluations. Third, the more thoroughly managers understand the information they receive, the more insight they gain into the factors affecting credit default, thus allowing them to combine statistical scores and expert judgement to make proper credit decisions. While various techniques have been applied to develop more comprehensible credit risk models, very few have focused on balancing accuracy and comprehensibility, which is required for a more exhaustive decision-making process.

Although increasingly complex models for the assessment of credit risk continue to be developed, these are not empirically useful because professionals in the financial industry primarily need comprehensible models that can be easily used in practice [27].

On the other hand, rule extraction techniques generate classification models that have clear advantages. First, they are comprehensible and can therefore be easily incorporated into financial applications where the classifications need to be extremely clear. Second, extracted rules only sacrifice a small degree of accuracy compared with the black box models from which they are generated [9].

1.6. Related works

Setiono and Liu [19] proposed NeuroLinear, a system for extracting oblique decision rules from NNs that have been trained for classification of patterns. Each condition of an oblique decision rule corresponds to a partition of the attributes' instance space by a hyperplane that is not necessarily axis-parallel. The novel algorithm Re-RX, originally intended as a rule extraction tool, was recently developed by Setiono et al. [22]. This algorithm provides a hierarchical, recursive consideration of discrete variables prior to analysis of continuous data, and is able to generate classification rules from NNs that have been trained on the basis of discrete and continuous attributes. Another algorithm for obtaining classification rules from discrete data, termed GRG (greedy rule generation), was written by Odajima et al. [28]. "Greedy" is included in the title because in each iteration, the algorithm attempts to find the optimum rule,

taking sample numbers and subspace sizes into consideration, as well as the quantity of attributes that the rule will contain. In this program, a standard decompositional approach is used to extract rules from NNs, such that NNs having a single hidden layer are trained, after which, the GRG algorithm considers the discretized hidden unit activation values.

Another recent rule extraction rule algorithm is Minerva [29], which is somewhat unique because it is applicable under an extremely wide range of circumstances. Minerva can be applied to regression and classification scenarios with both numerical and categorical data without invoking an underlying black box assumption.

Three quantized SVMs (QSVMs) have recently been proposed, representing unique DIMLP networks that are trained by employing an SVM learning algorithm [30] and applied for the purpose of rule extraction.

1.7. The Re-RX algorithm family

From system engineering perspective, the Re-RX algorithm cascade repeats the BPNN, the pruning, and C4.5 in a recursive cascade ensemble.

A major advantage of the Re-RX algorithm recently developed by Setiono et al. [22] is that it was designed as a rule extraction tool. It provides a hierarchical, recursive consideration of discrete variables prior to analysis of continuous data. In addition, it is capable of generating classification rules from NNs that have been trained on the basis of discrete and continuous attributes.

In other words, the Re-RX algorithm achieves highly accurate rule extraction and offers good comprehensibility through the generation of perfect or strict separation between discrete and continuous attributes in the antecedent of each extracted rule.

Recently, we proposed using both discrete and continuous attributes to generate the decision tree in the Re-RX algorithm framework (hereafter Continuous Re-RX) [22]. Although this seems to be counterintuitive with the design concept of the Re-RX algorithm, in that it results in the generation of a more complex decision tree, the use of both types of attributes is done to enhance accuracy [31,32].

To achieve both concise and highly accurate extracted rules while simultaneously maintaining the good framework of the Re-RX algorithm, we recently proposed supplementing the Re-RX algorithm with J48graft, a class for generating a grafted C4.5 decision tree (hereafter Re-RX with J48graft) [33,34].

Using the Re-RX algorithm, rules have been extracted from pruned NNs previously trained using all available data samples as well as more limited datasets [35,36]. Interestingly, there is little difference in the accuracy of predictions resulting from rule sets generated by pruned NNs trained using a selection of samples and the predictions made by the same program applying an NN trained using the complete dataset. This approach is deemed the "Sampling Re-RX" method.

Herein we provide the first comprehensive comparison of the results obtained by 10 runs of 10-fold cross validation (CV) of the Re-RX algorithm family based on the assessment of credit risk calculated from four real-life, two-class mixed credit-risk datasets. The results of the following methods are compared: the Re-RX algorithm [22]; Continuous Re-RX [31,32]; Re-RX with J48graft [33,34]; Sampling Re-RX [35,36]; NeuroLinear [19]; NeuroLinear+GRG [28]; and three unique rule extraction techniques involving SVMs [30] and Minerva [29].

We describe the Re-RX algorithm in Section 2.1, Continuous Re-RX in Section 2.2, Re-RX with J48graft in Section 2.5, and Sampling Re-RX in Section 2.7. In Section 3, we describe experimental datasets and setup, while in Section 4, we present the results. In Section 5, we discuss the experimental results and provide a detailed discussion on the Re-RX algorithm family from a Pareto optimal perspective, Continuous Re-RX vs. high performance classifiers, Re-RX with J48graft and Sampling Re-RX for the comprehensibility of extracted rules, and the potential of the Re-RX algorithm family for credit scoring with Big Data. Finally, in Section 6, we summarize our conclusions.

2. Method

2.1. Recursive rule extraction algorithm (Re-RX algorithm)

Although the Re-RX algorithm can easily handle multi-group problems, it was originally developed to consider only two-group classification problems [22]. The outline of the Re-RX algorithm is as follows:

Algorithm Re-RX (S, D, C)
Input: A set of data samples S having discrete attributes D and continuous attributes C.
Output: A set of classification rules.
1. Train and prune [37] an NN by using the dataset S and all of its D and C attributes.
2. Let D' and C' be the sets of discrete and continuous attributes, respectively, still present in the network, and let S' be the set of data samples correctly classified by the pruned network.
3. If $D'=\phi$, then generate a hyperplane to split the samples in S' according to the values of the continuous attributes C', and then stop.
Otherwise, use only the discrete attributes D' to generate the set of classification rules R for dataset S'.
4. For each rule, R_i is generated:
If support(R_i)>δ_1 and error(R_i)>δ_2, then
- Let S_i be the set of data samples that satisfy the condition of rule R_i and D_i be the set of discrete attributes that do not appear in rule condition R_i.
- If $D_i=\phi$, then generate a hyperplane to split the samples in S_i according to the values of their continuous attributes C_i, and then stop.
- Otherwise, call Re-RX (S_i, D_i, C_i).

Any NN training and pruning method can be used in Step 1 of the Re-RX algorithm, as it does not make any assumptions regarding the NN architecture; however, we have restricted ourselves to the use of BPNNs with only one hidden layer because such networks have been shown to retain the universal approximation property [38].

A crucial component of any NN rule extraction algorithm is an effective NN pruning algorithm. Pruning the inputs that are not needed to solve the problem allows the extracted rule set to be more concise, and a pruned network also helps to filter noise that might be present in the data, such as that from outlying or incorrectly labeled data samples. Therefore, from Step 2 onward, the algorithm only processes training data samples that have been correctly classified by the pruned network. Previously, we developed an NN pruning algorithm that incorporates a penalty function during training and adds a positive penalty value to the sum-of-squared error function for each connection with nonzero weight [37]. Consequently, many of the connections have weights very close to zero when network training is complete, and those with very small values can typically be pruned without adversely affecting the accuracy of the network.

If all discrete attributes are pruned from the network, the algorithm generates a hyperplane in Step 3

$$\sum_{Ci \in C'} wiCi = w0$$

that separates both groups of samples. Statistical and machine learning methods such as logit regression or SVMs can then be used to obtain the constant and the rest of the coefficients of the hyperplane. We employ an NN with one hidden unit in our implementation.

A set of classification rules comprising only discrete attributes is generated when at least one discrete attribute remains in the

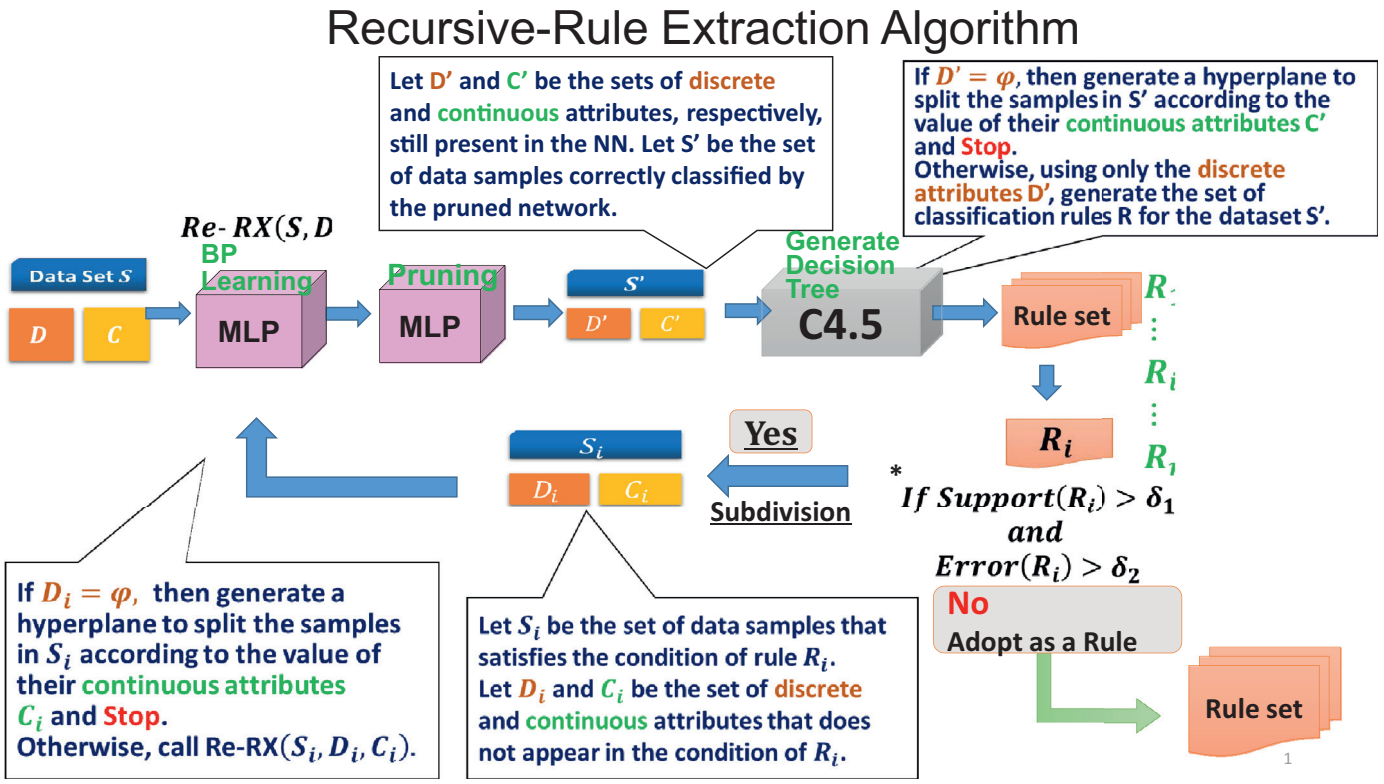

Fig. 1. Schematic overview of the Recursive-Rule eXtraction (Re-RX) algorithm.

pruned network, which effectively partitions the input space into smaller subspaces based on the values of the discrete attributes. Each subspace corresponds to a generated rule, and when the rule is not sufficiently accurate, the Re-RX algorithm is used to further partition the subspace.

The support of a rule, which is the percentage of samples covered by that rule, and each rule's corresponding error rate are checked in Step 4. If the support meets the minimum threshold δ_1 and the error rate exceeds the threshold δ_2, then the subspace of the rule is further subdivided either by calling Re-RX recursively when no discrete attributes remain present in the conditions of the rule, or by generating a separating hyperplane involving only the continuous attributes. Because the Re-RX algorithm handles discrete and continuous attributes separately, it generates a set of classification rules that are more comprehensible than those with both types of attributes in their conditions.

To enable a better understanding of its underlying mechanisms, a brief overview of the Re-RX algorithm and the concept behind its design is shown in Fig. 1. C4.5 [39] was used to generate decision trees in the Re-RX algorithm. The subdivision of the Re-RX algorithm is a unique function that is inherent in its nature. Each successive subdivision allows the use of other previously unused attributes; this increases the number of extracted rules as well as their accuracy.

It should be noted that the accuracy, comprehensibility, and conciseness of extracted rules have important trade-offs. Before subdivision, extracted rules are more comprehensible and concise, yet less accurate. Conversely, after subdivision, extracted rules are less concise, yet more accurate.

2.2. Re-RX algorithm with continuous attributes (Continuous Re-RX)

Although a primary aim of the Re-RX algorithm is the strict separation of discrete and continuous attributes in the antecedent of each extracted rule, this design often results in reduced accuracy. Whereas the Re-RX algorithm prunes continuous attributes (C') before the C4.5 decision tree is generated (Fig. 2), Continuous Re-RX uses both discrete (D') and continuous attributes (C') to generate the decision tree [31,32], which results in increased complexity. This may seem counterintuitive to the algorithm's design, but the use of both types of attributes also results in increased accuracy. An outline of Continuous Re-RX is as follows:

Continuous Re-RX (S′, D′, C′)
Input: A set of data samples (S′) having both discrete (D′) and continuous (C′) attributes.
Output: A set of classification rules.

1. Train and prune [37] an NN using the dataset S and all of its D and C attributes.
2. Let D′ and C′ be the sets of discrete and continuous attributes, respectively, still present in the network, and let S′ be the set of data samples correctly classified by the pruned network.
3. Generate decision tree by using both discrete (D′) and continuous (C′) attributes [31,32].
4. For each rule, R_i is generated:
   If support (R_i)>δ_1 and error (R_i)>δ_2, then
   - Let S_i be the set of data samples that satisfies the condition of rule R_i, let D_i be the set of discrete attributes, and let C_i be the set of continuous attributes that does not appear in rule condition R_i.
   - Call Continuous Re-RX (S_i, D_i, C_i).

   Otherwise, Stop.

As shown in Fig. 2, in Continuous Re-RX, we carefully set the value of the values of δ_1 and δ_2 in Step 4.

2.3. J4.8

J4.8 [40] is a Java-based version of C4.5 [39], which itself is an improved version of Quinlan's ID3 algorithm [41]. The decision trees generated by C4.5 are used for classification, so this algorithm is usually described as a statistical classifier. Although these algorithms are quite similar, the improvements C4.5 has over ID3 are that it uses the gain ratio to determine the best target attribute,

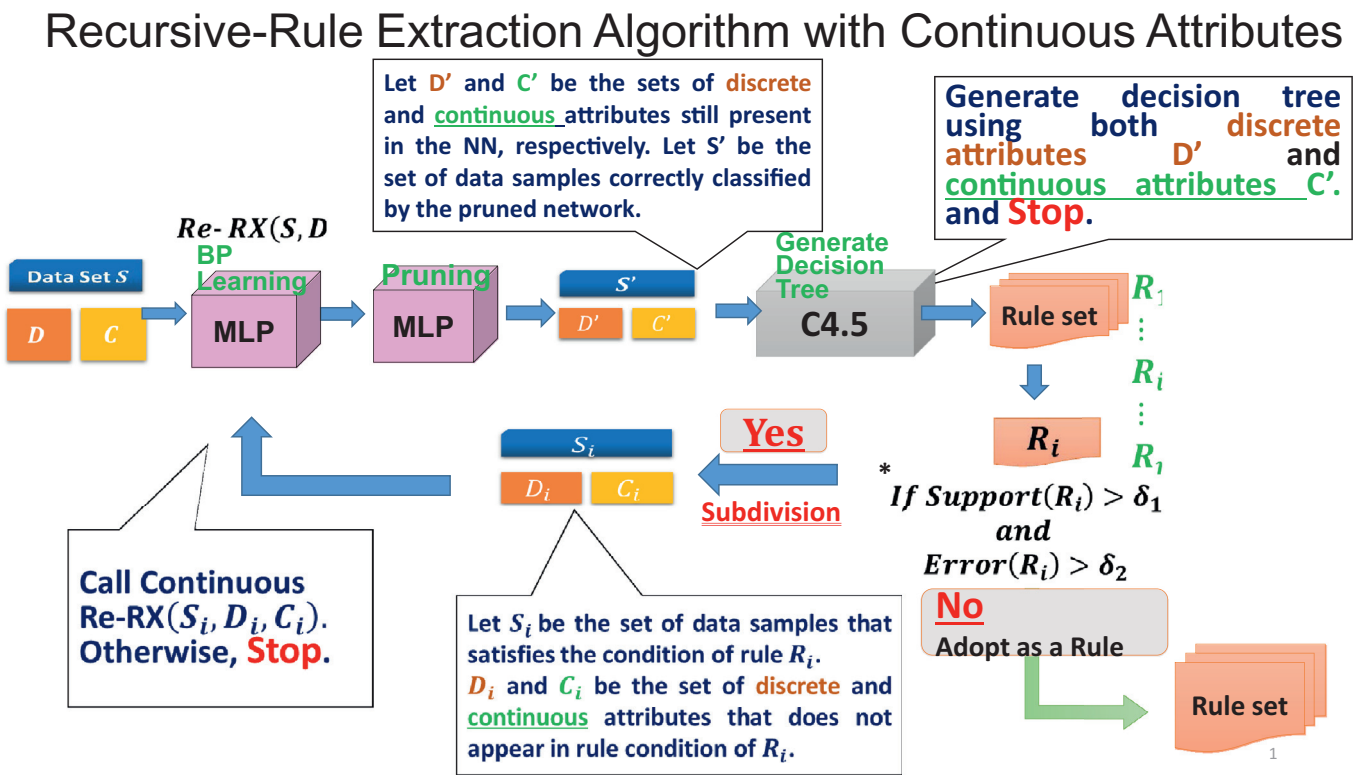

Fig. 2. Schematic overview of the Recursive-Rule eXtraction (Re-RX) algorithm with continuous attributes (Continuous Re-RX).

and, for numerical attributes, it creates a threshold and then splits the data into those whose attribute value is either greater, or less than or equal to, that threshold. C4.5 can also handle attributes with variable cost, and can prune the decision tree after its creation, which reduces its size and saves both time and memory.

2.4. J48graft

Decision tree grafting was developed in order to improve upon the "simplest is best" method for selecting a good tree. The basic tenet of tree grafting is that similar objects tend to have a high probability of belonging to the same class, and if following this process results in a better classification model, then yielding more complex trees becomes unnecessary.

Grafting is a post-process that can easily be applied to decision trees. The primary objective of grafting is to reclassify regions of an instance space containing no training data or only misclassified data, which in turn decreases prediction error. First, grafting identifies the best-suited cuts of existing leaf regions. Next, new leaves with classifications differing from the original are created via a branching out process, which increases the complexity of the tree naturally. However, tree grafting only considers branches that do not introduce classification errors in the data that has already been classified correctly, which ensures error reduction.

The C4.5A algorithm introduced by Webb, which is also referred to as the "all-tests-but-one partition (ATBOP)," is an even more efficient method for evaluating potentially supporting evidence [42]. It was the implementation of the C4.5A algorithm in open source data mining software (the Waikato Environment for Knowledge Analysis [Weka]) that led to the development of J48graft [40].

Pruning aims to reduce the complexity of a decision tree while retaining good predictive accuracy, and therefore can be thought of as the opposite of grafting. Despite these contrasts, or possibly because of them, Webb [43] concluded that pruning and grafting work well in parallel.

2.5. Re-RX algorithm with J48graft (Re-RX with J48graft)

With the objective of extracting more accurate and concise classification rules, we proposed replacing the conventional Re-RX algorithm, which uses C4.5 as a decision tree [39], with Re-RX with J48graft [33,34]. The conventional pruning used in J4.8 both complements and contrasts that used in J48graft [44]. The performance of the Re-RX algorithm [22] is thought to be greatly affected by the decision tree. To extract more accurate and concise classification rules, in consideration of the grafting concepts associated with J48graft, we decided to replace J4.8 with J48graft in the Re-RX algorithm.

We frequently employ Re-RX with J48graft [33,34] to form decision trees in a recursive manner while training MLPs using BP, which allows pruning [37] and therefore generates more efficient MLPs for rule extraction. A schematic overview of Re-RX with J48graft is shown in Fig. 3.

2.6. Sampling selection

In contrast to the development of more complex models for two-class classification problems such as credit scoring, Setiono [35,36] proposed a supervised learning scheme that aims to increase model accuracy by selecting the most appropriate training data samples.

In this scheme, models for classification problems, such as NNs, are trained using a historical dataset. In the case of classification problems such as credit scoring, the credit risk of each sample is labeled as either good or bad. However, some of these class labels may be incorrectly assigned, resulting in the presence of irregular data samples. Although these samples may have similar attributes, as is commonly the case for most samples in one class, they actually belong to a different class. This is problematic because the presence of irregular and/or mislabeled data samples in a training dataset is likely to adversely affect the performance of the NN.

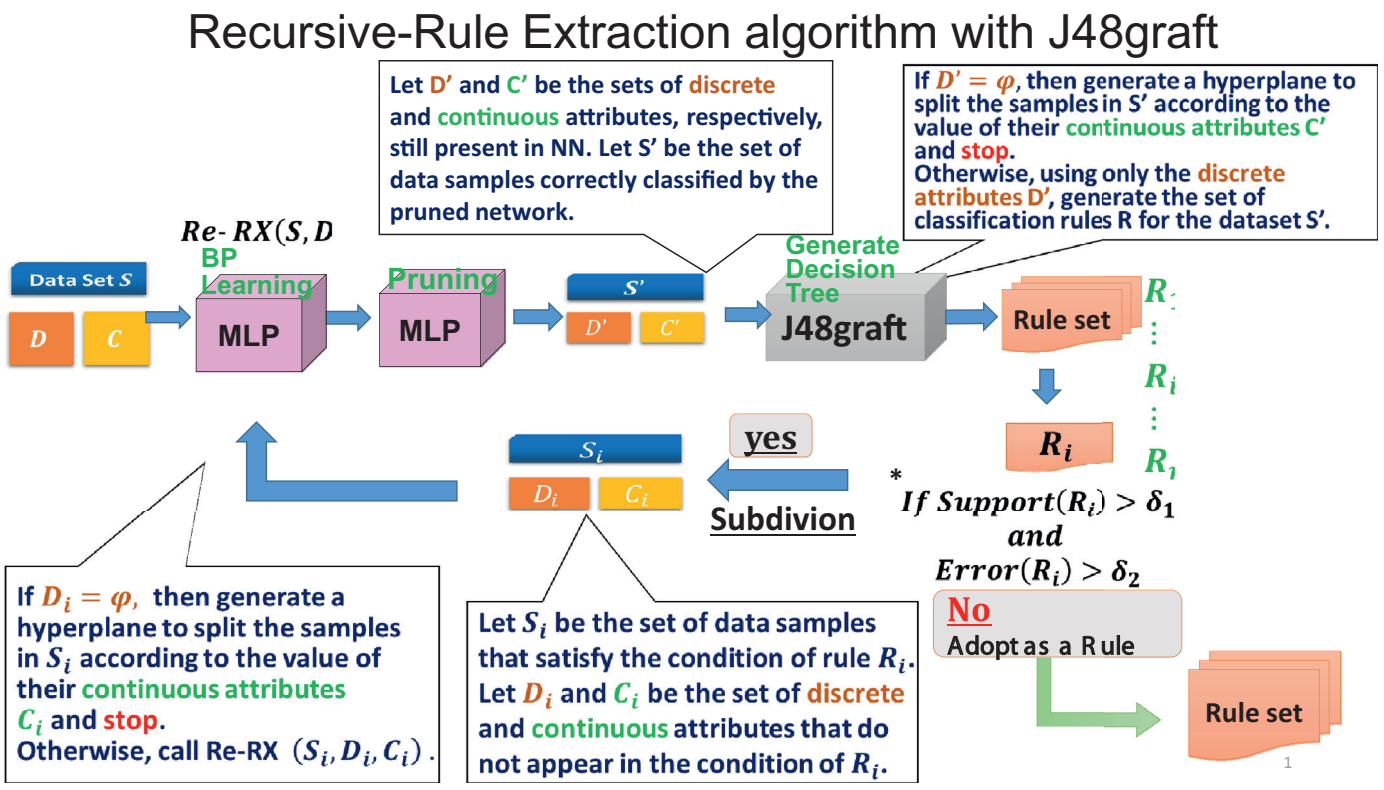

Fig. 3. Schematic overview of the Recursive-Rule eXtraction (Re-RX) algorithm with J48graft (Re-RX with J48graft).

In the sampling selection technique proposed by Setiono et al. [35,36], NNs are trained to identify potentially irregular and/or mislabeled data samples. Data samples that are consistently misclassified by a majority of NNs are then removed before a model is constructed to distinguish between good and bad credit risk.

The sampling selection technique can be summarized as follows: (1) Ensemble creation: train an ensemble of M feedforward NNs using the available training data samples; (2) Sample selection: select training data samples based on the predictions of the NN ensemble; (3) Model generation: use the selected samples to train an NN; and (4) Rule extraction: apply an NN rule extraction algorithm to obtain concise and interpretable classification rules capable of distinguishing between good and bad credits.

The selection of samples in Step 2 is, as the name suggests, a core component of the sampling selection technique. First, we employed an NN ensemble to identify outliers in the training dataset. An effective method for improving the predictive accuracy of numerous learning methods is to remove outliers and noise prior to learning. If a data sample is incorrectly classified by a proportion of NNs exceeding the threshold ρ and thereby identified as an outlier, it is discarded; otherwise, it is retained in the training dataset.

2.7. Re-RX algorithm combined with sampling selection technique (Sampling Re-RX)

Here we describe Re-RX combined with sampling selection techniques (Sampling Re-RX) for preprocessing.

The objective of this algorithm is to achieve highly accurate, concise, and interpretable classification rules for the credit scoring dataset. However, the credit scoring dataset for rule extraction was a financial dataset, so the focus was on decreasing the number of extracted rules and the average number of antecedents. To extract concise rules, we employed Sampling Re-RX, which is better suited for achieving concise and interpretable, as opposed to accurate, classificiation rules.

We preprocessed the credit scoring dataset using the sampling selection technique [35,36] to extract a fewer number of rules and a lower average number of antecedents. We then employed Sampling Re-RX to extract a set of concise and interpretable diagnostic rules. A schematic overview of Sampling Re-RX is shown in Fig. 4. As shown in the figure, in Sampling Re-RX, a supplementary cross-validation loop is carried out with sampling selection by an NN ensemble.

The most important objective of Sampling Re-RX in terms of credit scoring is to improve the conciseness and interpretability of extracted rules for financial professionals. Hereafter, Re-RX, Continuous Re-RX, Re-RX with J48graft and Sampling Re-RX are referred to as the "Re-RX algorithm family."

3. Datasets and experimental setup

Assigning accurate credit scores to consumers is a vital function of financial institutions. Credit scores are typically calculated using a mathematical decision model that establishes risk based on the assessment of various attributes such as the consumer's age and annual income. This assessment process must be transparent, so credit scores must be generated using a "white box" model.

To highlight both the effectiveness and the appropriateness of our proposed model for assessing credit risk, we used the following four real-life, two-class mixed credit-risk datasets: German, Australian, Bene1 and Bene2. A brief description of these datasets is provided below. These four datasets contain a wide variety of attributes, including continuous variables, nominal variables with a limited number of values, and nominal variables with a large number of values, and are therefore frequently referred to in the literature and utilized by financial researchers.

3.1. Australian credit dataset

Available through the University of California Irvine (UCI) Machine Learning Repository [45], the Australian Credit Dataset

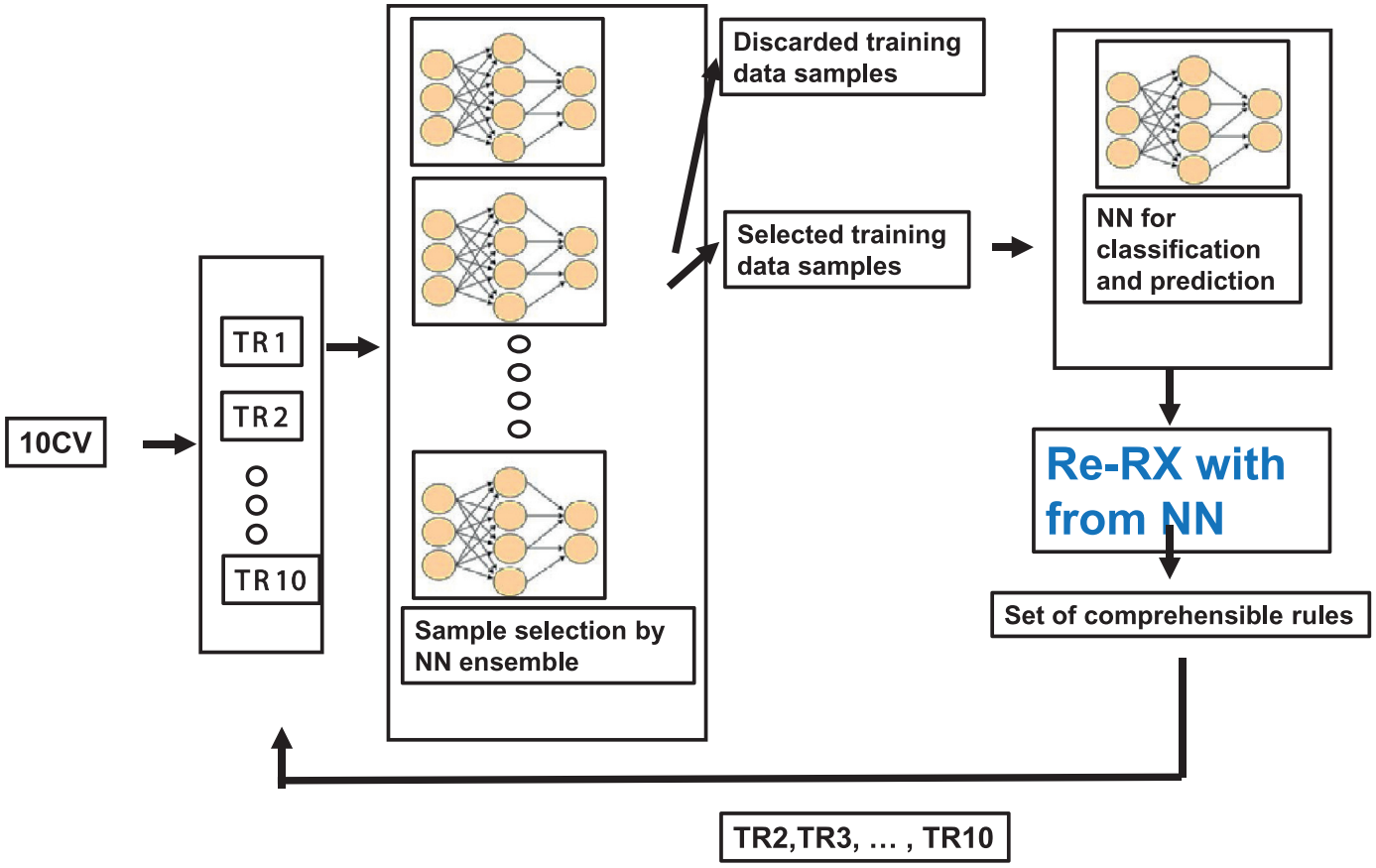

Fig. 4. Schematic overview of the Re-RX algorithm combined with sampling selection technique (Sampling Re-RX).

contains 690 samples with 14 attributes, eight discrete with two to 14 values, and six continuous. This dataset also contains 307 (approximately 44.5%) positive and 383 (approximately 55.5%) negative instances. To protect the confidentiality of the data, all attribute names and values have been changed to meaningless symbols. For the purposes of this study, we randomly divided this dataset into 50% training data and 50% test data.

3.2. German credit dataset

Also available through the UCI Machine Learning Repository is the German Credit Dataset [45], which contains 1000 samples with 20 attributes. Credit is classified as either good (about 700 samples) or bad (about 300 samples). Other attributes include: (1) current status of existing checking account; (2) duration of account in months; (3) credit history; (4) credit purpose; (5) credit line; (6) status of savings account(s)/bonds; (7) length of current employment; (8) installment rate in percentage of disposable income; (9) sex and marital status; (10) debtors/guarantors; (11) number of years in current residence; (12) property owned; (13) age; (14) other existing installment plans; (15) type of residence; (16) number of existing credit lines; (17) occupation; (18) current status regarding telephone service; (19) foreign worker status; and (20) number of dependents. For the purposes of this study, we randomly divided this dataset into 70% training data and 30% test data.

3.3. Bene1 and Bene2 credit datasets

In this study, the Bene1 and Bene2 datasets [9] used by Benelux-based major financial institutions to summarize consumer credit data were also used. In accordance with standard banking practices, customers in these voluminous datasets are flagged as high risk if they have ever been in payment arrears for more than 90 days. For the purposes of this study, we randomly divided these datasets into approximately 67% training data and 33% test data.

Table 1
Characteristics of datasets in credit risk evaluation.

	Dataset size	Input total	Input continuous	Input discrete
Australian	690	14	6	8
German	1000	20	7	13
Bene1	3123	27	18	9
Bene2	7190	28	18	19

3.4. Experimental setup

Next, the training sets within each database were used to train NNs and extract classification rules. One input unit was created for each continuous attribute in the dataset, and either thermometer or dummy variable encoding was used to convert discrete attributes into a binary input string [46]. The characteristics of the datasets used for evaluating credit risk are summarized in Table 1. In order to deal with the class imbalance problem commonly associated with credit scoring datasets, we used the area under the receiver operating characteristic curve (AUC-ROC) to evaluate performance because it does not include class distribution or misclassification costs [47].

4. Results

4.1. Performance

In order to guarantee the validity of the results, we used 10 runs of 10-fold CV [48] to evaluate the classification rule accuracy

Table 2
Comparison of results from Minerva, QSVM-L, QSVM-P3, QSVM-G, Re-RX*, Sampling Re-RX, Re-RX with J48graft and Continuous Re-RX for the German dataset (10 runs of 10-fold cross validation).

Index	Minerva	QSVM-L	QSVM-P3	QSVM-G	Re-RX*	Sampling Re-RX	Re-RX with J48graft	Continuous Re-RX
TS ACC %	70.51 ± 2.66	74.8 ± 0.8	75.10 ± 0.9	73.0 ± 1.3	71.82 ± 1.17	73.2 ± 0.32	72.78 ± 0.87	**75.22 ± 0.33**
AUC					0.65	0.66	0.650	0.692
# rules	**8.4**	77.4	85.4	170.6	45.1	19.34	16.65	39.6
Ave # Ante.	5.61	5.3	5.4	5.7	9.39	6.2	6.19	9.13

Re-RX: Recursive-Rule eXtraction; TS: testing dataset; AUC: area under the receiver operating characteristic (ROC) curve; # rules: number of rules; ACC: accuracy; Ave. # ante.: average number of antecedents; 10CV: 10-fold cross validation: 10×10CV: 10 runs of 10-fold cross validation; SVM: support vector machine.

Table 3
Comparison of results from the Minerva, NeuroLinear, NeuroLinear+GRG, QSVM-L, QSVM-P3, QSVM-G, Re-RX*, Sampling Re-RX, Re-RX with J48graft and Continuous Re-RX for the Australian dataset (10 runs of 10-fold cross validation).

Index	Minerva	Neuro Linear	Neuro Linear+GRG	QSVM-L	QSVM-P3	QSVM-G	Re-RX*	Sampling Re-RX	Re-RX with J48graft	Continuous Re-RX
TS ACC %	85.57 ± 1.70	83.64	86.40	85.60 ± 0.20	85.7 ± 0.6	85.6 ± 0.3	86.06 ± 0.36	86.48 ± 0.26	86.04 ± 0.29	**86.93 ± 0.29**
AUC							0.86	0.864	0.862	0.869
# rules	4.80	6.60	2.80	**2.0**	20.5	8.3	15.43	11.04	4.58	14.0
Ave # Ante.	2.93			1.0	3.7	2.6	6.23	5.27	2.38	5.95

Re-RX: Recursive-Rule eXtraction; TS: testing dataset; AUC: area under the receiver operating characteristic (ROC) curve; # rules: number of rules; ACC: accuracy; Ave. # ante.: average number of antecedents; 10CV: 10-fold cross validation: 10×10CV: 10 runs of 10-fold cross validation; SVM: support vector machine.

Table 4
Comparisons between the Re-RX algorithm family for the Bene1 dataset (10 runs of 10-fold cross validation).

Index	Re-RX*	Sampling Re-RX	Re-RX with J48graft	Continuous Re-RX
TS ACC %	70.22 ± 1.93	72.06 ± 0.09	70.95 ± 0.37	**72.50 ± 0.35**
AUC	0.679	0.68	0.669	0.702
# rules	43.2	**25.46**	27.74	48.40
Ave. # Antecedents	7.57	6.25	7.38	7.52

Re-RX: Recursive-Rule eXtraction; TS: testing dataset; AUC: area under the receiver operating characteristic (ROC) curve; # rules: number of rules; ACC: accuracy; Ave. # ante.: average number of antecedents; 10CV: 10-fold cross validation: 10×10CV: 10 runs of 10-fold cross validation; SVM: support vector machine.

Table 5
Comparisons between the Re-RX algorithm family for the Bene2 dataset (10 runs of 10-fold cross validation).

Index	Re-RX*	Sampling Re-RX	Re-RX with J48graft	Continuous Re-RX
TS ACC %	71.66 ± 0.56	72.6 ± 0.12	70.69 ± 0.51	**74.67 ± 0.58**
AUC	0.616	0.60	0.615	0.648
# rules	54.4	28.31	**27.61**	75.9
Ave. # Antecedents	7.81	6.1	6.46	7.95

Re-RX: Recursive-Rule eXtraction; TS: testing dataset; AUC: area under the receiver operating characteristic (ROC) curve; # rules: number of rules; ACC: accuracy; Ave. # ante.: average number of antecedents; 10CV: 10-fold cross validation: 10×10CV: 10 runs of 10-fold cross validation; SVM: support vector machine.

of test datasets. The k-fold CV method is widely applied by researchers to minimize the bias associated with random sampling.

We trained the credit scoring datasets using the Re-RX algorithm family and obtained 10 runs of 10-fold CV accuracies for the test dataset (TS ACC), the number of extracted rules (# rules), the average number of antecedents (Ave. # ante.), and the AUC-ROC.

Numerous types of rules have been suggested in the literature from the perspective of the expressive power of extracted rules, including propositional rules, which take the form of IF-THEN expressions and clauses defined using propositional logic, and *M*-of-*N* rules. Breaking from traditional logic, fuzzy rules allow partial truths instead of Boolean true/false outcomes.

Even if all types of rules are considered, the consensus is that no matter how they are defined, an ideal measure has yet to be developed; therefore, "what is a concise and/or interpretable rule?" remains a difficult question to answer. To address this, we attempted to develop a "rough indicator" of conciseness by comparing the average number of antecedents from extracted rules generated using a variety of techniques.

In Tables 2 through 5, we conducted the first comprehensive performance comparisons of Minerva [29], NeuroLinear [19], NeuroLinear+GRG [28], three rule extraction techniques using SVMs [30], and the Re-RX algorithm family for each of the four credit scoring datasets.

Some results displayed in Tables 2 and 3 were obtained from previously published literature [19,28–30]. All values for other methods in the tables were generated using 10 runs of 10-fold CV. Since use of 10-fold CV was not clearly described in the original literature, we independently implemented the Re-RX algorithm, i.e., Re-RX, from the original authors [22], and obtained accuracies using 10 runs of 10-fold CV.

5. Discussion

5.1. Performance for the German dataset

Continuous Re-RX showed the best accuracy for the test dataset, nearly identical to that obtained by QSVM-P3 [30] and QSVM-L [30] with a much higher number of extracted rules. Minerva provided the smallest number of extracted rules. The number of extracted rules obtained using Re-RX* was less than half that of Sampling Re-RX and Re-RX with J48graft. Although QSVM-L, QSVM-P3, and QSVM-G [30] showed better accuracy, they also had a much higher number of extracted rules.

5.2. Performance for the Australian dataset

Continuous Re-RX showed the best test accuracy. However, the test accuracies for all algorithms were between 85.5% and 86.93%. On the other hand, QSVM-L showed a remarkably concise number of rules (2.0) with only 1.0 antecedent. However, considering the reduced conciseness for the German Dataset (Table 2), which was obtained using the same method, we should not presuppose such excellent conciseness for all types of datasets.

NeuroLinear+GRG also showed a concise number of rules at 2.8. Comparing NeuroLinear+GRG with NeuroLinear, GRG preprocessing for the Australian Dataset was very effective for conciseness and accuracy.

The number of rules and the average number of antecedents obtained by NeuroLinear for the German Dataset [19] were 2.0

Table 6
Performance of various classifiers and Continuous Re-RX, Sampling Re-RX, and Re-RX with J48graft for the Australian dataset.

Method [10×10CV]	TS ACC (%)	# Rules	Rule set	Ave. # ante.	Year [ref.]
Vertical bagging decision trees model	91.97	–	–	–	2010 [49]
Weighted-least squares SVM_{RBF}	90.63	–	–	–	2011 [50]
Kernel, fuzzification, penalty factors-multi-criteria optimization classifier	88.84	–	–	–	2014 [51]
Weighted-case-based-reasoning with preference functions optimized with GA	88.55	–	–	–	2012 [52]
Random space bagging decision tree	88.01	–	–	–	2012 [53]
Neighborhood + rough set + SVM_{RBF}	87.52 ± 0.052	–	–	—	2011 [54]
Decision tree ensemble (boosting-100)	87.23	–	–	—	2014 [55]
Hidden Markov model/group method of data handling	87.02 ± 1.56	–	–	––	2013 [56]
Continuous Re-RX	**86.93 ± 0.29**	14	Possible	5.95	2016[31,32]
SVM + stratified sampling	86.83 ± 3.96	–	–	–	2012 [57]
Sampling Re-RX	86.48 ± 0.26	11.04	Possible	5.27	2015 [35,36]
Artificial immune network-based classifier	86.38	–	–	–	2012 [58]
Axiomatic fuzzy set (5CV)	86.22	451.6	–	–	2013 [59]
Naïve Bayes + wrapper (GA) method	86.09	–	–	–	2015 [60]
Re-RX J48graft	86.04 ± 0.87	**4.58**	**Yes**	**2.38**	2016 [33,34]
Group-wise feature selection (50×5CV)	85.6 ± 2.5	–	–	–	2012 [61]

Re-RX: Recursive-Rule eXtraction; TS: testing dataset; ACC: accuracy; Ave. # ante.: average number of antecedents; 10CV: 10-fold cross validation: 5CV: 5-fold cross validation: 10×10CV: 10 runs of 10-fold cross validation; 50×5CV: 50 runs of 5-fold cross validation; SVM: support vector machine; RBF: radial basis function; GA: genetic algorithm.

and 8.5, respectively [28]. Although the number of extracted rules was quite small, the number of antecedents was quite high. Although NeuroLinear+GRG did not provide an average number of antecedents for the Australian Dataset, it may show a tendency similar to NeuroLinear in that the average number of antecedents would not be very interpretable. Minerva and Re-RX with J48graft showed a considerably smaller number of extracted rules than Re-RX*.

5.3. Performance for the Bene1 dataset

All methods in the Re-RX algorithm family showed approximately the same test accuracies.

In contrast, Sampling Re-RX and Re-RX with J48graft showed about a 30% reduction in the number of rules extracted using Re-RX*. Since the tendency to generate more rules than other rule extraction algorithms is the most serious problem in Re-RX, this result is quite meaningful.

5.4. Performance for the Bene2 dataset

Continuous Re-RX showed the best test accuracy, but was only slightly better than the other methods. In addition, it showed a much higher number of extracted rules than Re-RX*. On the other hand, Sampling Re-RX and Re-RX with J48graft showed about the half number of rules extracted using Re-RX*. In the same manner as that in Section 5.3, this result was quite meaningful.

5.5. Performance of the Re-RX algorithm family from a Pareto optimal perspective

As the present experiments have demonstrated, no extraction methods that generate both highly accurate and comprehensible rules for credit scoring datasets have been identified. Therefore, as noted in Section 2.1, the best approach appears to be finding a good balance between the two. However, to the best of our knowledge, no ideal rule extraction algorithm has been reported. In the light of this situation, we decided to develop highly accurate rule extraction methods that also offered high comprehensibility by generating perfect or strict separation between discrete and continuous attributes in the antecedent of each extracted rule.

Obtaining a small number of extracted rules from a dataset does not guarantee that the rules extracted by another algorithm, such as when the Australian Dataset was processed by NeuroLinear+GRG, will have higher comprehensibility. To compare the extent of comprehensibility between different algorithms, the average number of antecedents per extracted rule is a good indicator.

Needless to say, if we can find a Pareto optimal solution, we will obtain the best rule extraction algorithm. Ideally, we hope to extend the Pareto optimal curve to obtain a wider viable region and provide improvements in both accuracy and comprehensibility. Several newly developed multi-objective optimization formulas using revolutionary computation and related-techniques could be used to provide a theoretical and practical basis for determining a good balance between accuracy and comprehensibility.

5.6. Continuous Re-RX vs. high performance classifiers

For further analysis, we tabulated the accuracies of high performance classifiers for the Australian Dataset, which were recently reported using 10 runs of 10-fold CV, as an example.

As shown in Table 6, the difference in classification accuracy obtained by Continuous Re-RX has been approaching within 5% of that obtained by the highest current performance classifier [49].

From the perspective of rule extraction, the number of rules for high performance classifiers, if applicable, can be treated as infinite. In other words, the only objective function of the classifier is its accuracy. Therefore, high performance classifiers are no different from resignation to find a compromise between both requirements by building a simple rule set that mimics how the well-performing complex model (black-box) makes it decisions.

Although the number of rules extracted using Continuous Re-RX was much higher than that of other algorithms in the Re-RX algorithm family, Continuous Re-RX still has the best rule extraction accuracy, with only slightly lower separation capability between discrete and continuous attributes in the antecedent of each extracted rule. The competition for achieving only better classification accuracy for the credit scoring dataset has appeared to plateau [49,50], and unless classification accuracy can be considerably improved, i.e., close to 100%, a very limited contribution will be made to the financial services industry.

5.7. Re-RX with J48graft and Sampling Re-RX for comprehensibility of extracted rules

Recently, Chen et al. [62] reported that financial rule extraction is completely algorithmic or automatic in most systems, and

has little supervision or user interaction. In order to acquire useful and comprehensible knowledge, users need to be integrated into a black box process through an interactive visual framework. However, in their study, Chen et al. ignored various rule extraction algorithms and/or methods.

In contrast, Fortuny and Martens [63] claimed that comprehensibility is required in any domain in which a model needs to be validated before it can be used in practice, such as medical diagnosis or audit mining.

In credit scoring, this requirement is a legal one [8], as described in Section 1.2. The Basel III Capital Accord includes similar requirements in relation to models for internal capital requirement calculations. Furthermore, findings from previous studies have shown that when the inner workings of a decision-making system are not understood by users, they will be skeptical and reluctant to use the model, even if it is well known to improve performance. Although the importance of comprehensibility has long been established [64], current data mining research seems to have a sole focus on predictive accuracy only. While it is possible to increase comprehensibility by constraining or modifying existing techniques (e.g., as in [65]), it is often more desirable to inspect the behavior of well-studied techniques without altering their inner workings. Rule extraction techniques have been proposed as a method to generate predictive rules that mimic the classifications made by the black-box technique without modifications [66], and they play an important role in data mining, which has been described as a process of finding novel and useful patterns in data [67].

As shown below, the rule set generated from the Australian Dataset using Re-RX with J48graft provides insight into the logic underlying the black-box model in human-readable form.

The extracted rule set obtained by the Re-RX with J48graft in the present paper is quite concise and intepretable for users. Since the Australian Dataset includes categorical attributes, A4, A5, A6, A12, we converted these into binary code as follows:

R1: D31 = 0 Then Class 1
R2: D28 = 0 AND D31 = 1 AND D32 = 0 Then Class 1
R3: D28 = 1 AND D31 = 1 AND D32 = 0 Then Class 2
R4: D31 = 1 AND D32 = 1 Then Class 2.

The average number of extracted rules was 4.0 and the average number of antecedents was only 1.75. Furthermore, only three attributes (D28, D31 and D32) were used. However, the predictive accuracy of the entire rule set was 86.04 ± 0.29. As shown in Table 6, comparing the classification accuracies, that obtained using Re-RX with J48graft (86.04 ± 0.29) was about 5.93% lower than that of the best performance classifier [49].

Certainly, recent high performance classifiers with a grand-scale techniques have shown very high classification accuracies; however, we believe that if the quality of the extracted rules from the financial datasets is strongly considered, the opportunity for supervision and interaction of financial professionals will be dramatically increased. Because transparency is necessary for datasets, using Re-RX with J48graft and Sampling Re-RX is expected to encourage and motivate new financial data analytics.

In fact, the Re-RX algorithm family has already been used for concise and interpretable extracted rules in regard to medical diagnosis for breast cancer [34] and thyroid diseases [32]. In these cases, the quality of rules is emphasized over the classification accuracy.

5.8. Potential of the Re-RX algorithm family for credit scoring with big data

To date, many studies have used both the German and Australian Datasets as part of a general tendency to employ either small- (below 1000 samples) or medium-sized (1000–10,000 samples) datasets [2]. However, more recently, the use of datasets with more than 10,000 samples has somewhat increased [68].

Finlay [1] demonstrated that the credit risk calculations of prominent financial institutions typically use datasets that are either small or of low dimensionality. The largest dataset used in the present study was the Bene2 Dataset, which contains 7190 samples, because the real-world datasets discussed by Finlay, which contain 88,789 and 138,606 samples [1], were considered too large to allow rule extraction via the Re-RX algorithm family in a reasonable time frame.

Regarding the complexity of the Re-RX algorithm family, Re-RX with J48graft took about 5 s to train the German Dataset using a standard workstation computer (3.1 GHz Intel Xeon E5-2687 W, 3.5 GHz Turbo, 25 MB Cache; 64 GB RAM; 512 GB DDR3 System memory). The testing time was negligible.

Therefore, presently, the Re-RX algorithm family remains difficlut to use in real-time and/or online tasks for large-scale credit scoring. However, with considerable improvement in information technology and processing speeds, the Re-RX algorithm family can be expected to run much faster on standard workstations or conventional personal computers.

6. Conclusion

In this study, we conducted the first comprehensive performance comparison based on 10 runs of 10-fold CV between the Re-RX algorithm family, NeuroLinear, NeuroLinear+GRG, Minerva and three rule extraction techniques from SVMs by applying these programs to four different real-life, two-class mixed credit-risk datasets.

The high accuracy of Continuous Re-RX was superior to that obtained using Re-RX, Re-RX with J48graft, Sampling Re-RX, NeuroLinear, NeuroLinear+GRG, Minerva and three SVM-based methods.

Re-RX with J48graft and Sampling Re-RX both use a recursive cascade ensemble to construct a unique hybrid classifier ensemble with perfect or strict separation between discrete and continuous attributes in the antecedents of extracted rules, so as to maintain high comprehensibility. Therefore, these two algorithms generated highly comprehensible rules with perfect or strict separation.

These findings suggest that Continuous Re-RX, Re-RX with J48graft, and Sampling Re-RX comprise a powerful management tool that allows the creation of advanced, accurate, concise and interpretable decision support systems for credit risk evaluation.

In addition, the superior features of the Re-RX algorithm family from the Pareto optimal perspective were discussed, as well as Continuous Re-RX vs. high performance classifiers, and Re-RX with J48graft and Sampling Re-RX in relation to the comprehensibility of extracted rules and the potential of the Re-RX algorithm family in credit scoring with Big Data.

In future studies, we intend to develop much more accurate and comprehensible rule extraction algorithms for large-sized datasets, and to attempt to come close to achieving true rule extraction from Big Data.

Acknowledgments

The author would like to express sincere appreciation to his graduate students, Atsushi Hara, Ryutaro Ono, Yuki Tanaka, Satoshi Nakano, Shota Fujisawa, and Tomoki Izawa, for their faithful efforts during this research.

References

[1] Finlay SM. Multiple classifier architectures and their applications to credit risk assessment. *Eur J Oper Res* 2011;**210**:368–78.

[2] García V, Marqués AI, Sánchez JS. An insight into the experimental design for credit risk and corporate bankruptcy prediction systems. *J Intell Inf Syst* 2015;**44**:159–89.
[3] Zhao Z, Xu S, Kang BH, Kabir MMJ, Liu Y. Investigation and improvement of multi-layer perceptron neural networks for credit scoring. *Expert Syst Appl* 2015;**42**:3508–16.
[4] The Federal Reserve Board. Federal Reserve Statistical Release G19. February 2015.
[5] Bank of England. Statistical Interactive Database: series LPMVZRF. February 2015.
[6] The Federal Reserve Board. Charge-off and Delinquency Rates. November 2014.
[7] Bank of England. Trends in Lending. January 2015.
[8] Martens D, Baesens B, Van Gestel T, Vanthienen J. Comprehensible credit scoring models using support vector machines. *Eur J Operat Res* 2007;**183**:1488–97.
[9] Baesens B, Setiono R, Mues C, Vanthienen J. Using neural network rule extraction and decision tables for credit-risk evaluation. *Manage Sci* 2004;**49**:312–29.
[10] Abellan J, Mantas C. Improving experimental studies about ensembles of classifiers for bankruptcy prediction and credit scoring. *Expert Syst Appl* 2014;**41**:3825–30.
[11] Finlay SM. Are we modelling the right thing? The impact of incorrect problem specification in credit scoring. *Expert Syst Appl* 2009;**36**:9065–71.
[12] Hansen LK, Salamon P. Neural network ensembles. *IEEE Trans Patter Anal Mach Intell* 1990;**12**:993–1001.
[13] Igelnik S, Pao YH, LeClair SR, Shen CY. The ensemble approach to neural-network learning and generalization. *IEEE Trans Neural Netw* 1999;**10**:19–30.
[14] Liao JJ, Shih C-H, Chen TF, Hsu MF. An example-based model for two-class imbalanced financial problem. *Econ Model* 2014;**37**:175–83.
[15] Braket N, Bradely AP. Rule extraction from support vector machine: a review. *Neurocomputing* 2010;**74**:178–90.
[16] Setiono R, Liu H. Symbolic representation of neural networks. *IEEE Comput* 1996;**29**:71–7.
[17] Craven M, Shavlik J. Extracting tree-structured representations of trained networks. In: *Advances in neural information processing systems (NIPS)*. Cambridge, MA: MIT Press; 1996. p. 24–30.
[18] Nauck D, Kruse R. Neuro-fuzzy methods in fuzzy rule generation. In: *Fuzzy sets in approximate reasoning and information systems*. Norwell, MA: Kluwer; 1999. p. 305–34.
[19] Setiono R, Liu H. Neurolinear: From neural networks to oblique decision rules. *Neurocomputing* 1997;**17**:1–24.
[20] Setiono R, Liu H. A connectionist approach to generating oblique decision trees. *IEEE Trans Syst Man Cybern B* 1999;**29**:440–4.
[21] Setiono R, Baesens B, Mues CA. A note on knowledge discovery using neural Setiono networks and its application to credit card screening. *Eur J Operat Res* 2009;**192**:326–32.
[22] Setiono R, Baesens B, Mues C. Recursive neural network rule extraction for data with mixed attributes. *IEEE Trans Neural Netw* 2008;**19**:299–307.
[23] Gupta A, Park S, Lam SM. Generalized analytic rule extraction for feedforward neural networks. *IEEE Trans Knowl Data Eng* 1999;**11**:985–91.
[24] Etchell TA, Lisboa JPG. Orthogonal search-based rule extraction (OSRE) for trained neural-networks: a practical and efficient approach. *IEEE Trans Neural Netw* 2006;**17**:374–84.
[25] Bologna G. A model for single and multiple knowledge based networks. *Artificial Intell Med* 2003;**28**:141–63.
[26] Zhou ZH. Extracting symbolic rules from trained neural network ensembles. *AI Commun* 2003;**16**:3–15.
[27] Florez-Lopez R, Ramon-Jeronimo JM. Enhancing accuracy and interpretability of ensemble strategies in credit risk assessment. A correlated-adjusted decsion forest proposal. *Expert Syst Appl* 2015;**42**:5737–53.
[28] Odajima K, Hayashi Y, Tianxia G, Setiono R. Greedy rule generation from discrete data and its use in neural network rule extraction. *Neural Netw* 2008;**21**:1020–8.
[29] Huysmans J, Setiono R, Baesens B, Vanthienen J. Minerva: sequential covering for rule extraction. *IEEE Trans Syst Man Cybern Part B* 2008;**38**:299–309.
[30] Bologna G, Hayashi Y. QSVM: a support vector machine for rule extraction. In: *IWANN 2015 Part II June 10-12*, 9095. Mallorca, Spain: LNCS; 2015. p. 276–89.
[31] Hayashi Y, Fujisawa S. Strategic approach for the multiple-MLP ensemble Re-RX algorithm. In: *Proceedings of international joint conference on neural networks (IJCNN 2015) July 12-17*; 2015. p. 669–76.
[32] Hayashi Y, Nakano S, Fujisawa S. Use of the recursive-rule extraction algorithm with continuous attributes to improve diagnostic accuracy in thyroid disease. *Inf Med Unlocked* 2016;**1**:1–8.
[33] Hayashi Y, Tanaka Y, Takagi T, Saito T, Iiduka H, Kikuchi H, Bologna G, Mitra S. Recursive-rule extraction algorithm with j48graft and applications to generating credit scores. *J Artificial Intell Soft Comput Res* 2016;**6**:35–44.
[34] Hayashi Y, Nakano S. Use of a Recursive-rule extraction algorithm with j48graft to archive highly accurate and concise rule extraction from a large breast cancer dataset. *Inf Med Unlocked* 2016;**1**:9–16.
[35] Setiono R. Sampling selection and neural network rule extraction for credit scoring. In: *Proceedings of the 43rd decision sciences institutes annual meeting*; 2012. p. 1280–90.
[36] Setiono R, Azcarraga A, Hayashi Y. Using sample selection to improve accuracy and simplicity of rules extracted from neural networks. *Int J Comp Intell Appl* 2015;**14**:1550021.
[37] Setiono R. A penalty-function approach for pruning feedforward neural networks. *Neural Comput* 1997;**9**:185–204.
[38] Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. *Neural Netw* 1989;**2**:359–66.
[39] Quinlan JR. Programs for machine learning. *Morgan Kaufmann series in machine learning*. San Mateo, CA: Morgan Kaufman, Inc.; 1993.
[40] Witten IH, Frank E. *Data mining: practical machine learning tools with java implementations*. San Mateo, CA: Morgan Kaufmann, Inc.; 1999.
[41] Quinlan JR. ID3: induction of decision trees. *Mach Learn* 1986;**1**:81–106.
[42] Webb GI. Decision tree grafting from the all-tests-but-one partition. In: *Proceedings of 16th international joint conference on artificial intelligence (IJCAI)*, 2; 1999. p. 702–7.
[43] Webb GI. Decision tree grafting. In: *Learining, IJCAI'97 proceedings of 15th international conference on artificial intelligence (IJCAI)*, 2; 1997. p. 846–85.
[44] http://fiji.sc/javadoc/weka/classifiers/trees/J48graft.html, (accssed 16.01.30).
[45] Frank A, Asuncion A. *Irvine machine learning repository*. University of California; 2010 *http://archive.ics.uci.edu/ml/*.
[46] Smith M. *Neural networks for statistical modeling*. New York: Van Nostrand Reinhold; 1993.
[47] Marqués AI, García V, Sánchez JS. On the suitability of resampling techniques for the class imbalance problem in credit scoring. *J Operat Res Soc* 2013;**64**:1060–70.
[48] Salzberg SL. On comparing classifiers: pitfalls to avoid and recommended approach. *Data Mining Knowl Discov* 1997;**1**:317–28.
[49] Zhang D, Zhou X, Leung SCH, Zheng J. Vertical bagging decision trees model for credit scoring. *Expert Syst Appl* 2010;**37**:7838–43.
[50] Yu L, Yao X, Wang S, Lai KK. Credit risk evaluation using a weighted least squares SVM classifier with design of experiment for parameter selection. *Expert Syst Appl* 2011;**38**:15392–9.
[51] Zhang Z, Gao G, Shi Y. Credit risk evaluation using multi-criteria optimization classifier with kernel, fuzzification and penalty factors. *Eur J Operat Res* 2014;**237**:335–48.
[52] Vukovic S, Delibasic B, Uzelac A, Suknovic M. A case-based reasoning model that uses preference theory functions for credit scoring. *Expert Syst Appl* 2012;**39**:8389–95.
[53] Wang G, Ma J, Huang L, Xu K. Two credit scoring models based on dual strategy ensemble trees. *Knowl Based Syst* 2012;**26**:61–8.
[54] Ping Y, Yongheng L. Neighborhood rough set and SVM based hybrid credit scoring classifier. *Expert Syst Appl* 2011;**38**:11300–4.
[55] Tsai C-F, Hsub Y-F, Yen DC. A comparative study of classifier ensembles for bankruptcy prediction. *Appl Soft Comput* 2014;**24**:977–84.
[56] Teng GE, He CZ, Xiao J, Jiang XY. Customer credit scoring based on HMM/GMDH hybrid model. *Knowl Inf Syst* 2013;**36**:731–47.
[57] Hens AB, Tiwari MK. Computational time reduction for credit scoring: an integrated approach based on support vector machine and stratified sampling method. *Expert Syst Appl* 2012;**39**:6774–81.
[58] Chang SY, Yeh TY. An artificial immune classifier for credit scoring analysis. *Appl Soft Comput* 2012;**12**:611–18.
[59] Liu X, Feng X, Pedrycz W. Extraction of fuzzy rules from fuzzy decision trees: an axiomatic fuzzy sets (AFS) approach. *Data Knowl Eng* 2013;**84**:1–25.
[60] Liang D, Tsai CF, Wu HT. The effect of feature selection on financial distress prediction. *Knowl Based Syst* 2015;**73**:289–97.
[61] Gönen GB, Gönen M, Gürgen F. Probabilistic and discriminative group–wise feature selection methods for credit risk analysis. *Expert Syst Appl* 2012;**39**:11709–17.
[62] Chen N, Ribeiro B, Chen A. Financial credit risk assessment: a recent review. *Artificial Intell Rev* 2016;**45**:1–23.
[63] Fortuny EJD, Martens D. Active learning-based pedagogical rule extraction. *IEEE Trans Neural Netw Learn Syst* 2015;**26**:2664–77.
[64] Kodratoff Y. The comprehensibility manifesto. *AI Commun* 1994;**7**:83–5.
[65] Chorowski J, Zurada JM. Learning understandable neural networks with nonnegative weight constraints. *IEEE Trans Neural Netw Learn Syst* 2015;**26**:62–9.
[66] Martens D, Gestel TV, Baesens B. Decompositional rule extraction from support vector machines by active learning. *IEEE Trans. Knowledge and Data Eng* 2008;**21**:178–91.
[67] Fayyad U, Piatetsky-Shapiro G, Smyth P. From data mining to knowledge discovery in databases. *AI Magine* 1996;**17**:37–54.
[68] Tsai CF, Hsu YF. A meta-learning framework for bankruptcy prediction. *J Forecast* 2013;**32**:167–79.