

A literature review on the application of evolutionary computing to credit scoring

August 3, 2012

Abstract

The last years have seen the development of many credit scoring models for assessing the creditworthiness of loan applicants. Traditional credit scoring methodology has involved the use of statistical and mathematical programming techniques such as discriminant analysis, linear and logistic regression, linear and quadratic programming, or decision trees. However, the importance of credit grant decisions for financial institutions has caused growing interest in using a variety of computational intelligence techniques. This paper concentrates on evolutionary computing, which is viewed as one of the most promising paradigms of computational intelligence. Taking into account the synergistic relationship between the communities of Economics and Computer Science, the aim of this paper is to summarize the most recent developments in the application of evolutionary algorithms to credit scoring by means of a thorough review of scientific articles published during the period 2000–2012.

Keywords: Credit scoring; Evolutionary computation; Genetic algorithms; Genetic programming; Classification; Variable selection; Parameter optimization

1 Introduction

Credit allows accessing to resources today with an agreement to repay over a period of time, usually at regular intervals. The resources may be financial, or they may consist of products or services. Credit has now turned into a very important

component in everyday life. Although credit cards are currently the most popular form of credit, other credit plans include residential mortgages, auto loans, student loans, small business loans, trade financing and bonds, among others.

The main risk for banks and financial institutions comes from the difficulty to distinguish the creditworthy applicants from those who will probably default on repayments. The recent world financial crisis has aroused increasing attention of financial institutions on credit risk prediction and assessment. The decision to grant credit to an applicant was traditionally based upon subjective judgments made by human experts, using past experiences and some guiding principles. Common practice was to consider the classic 3 C's, 4 C's or 5 C's of credit: character, capacity, capital, collateral and conditions (Abrahams and Zhang 2008). This method suffers, however, from high training costs, frequent incorrect decisions, and inconsistent decisions made by different experts for the same application.

These shortcomings have led to a rise in more formal and accurate methods to assess the risk of default. In this context, automatic credit scoring has become a primary tool for financial institutions to evaluate credit risk, improve cash flow, reduce possible risks and make managerial decisions (Thomas et al. 2002). Credit scoring is the set of decision models and their underlying methods that help lenders determine whether credit should be approved to an applicant. The ultimate goal of credit scoring is to assess credit worthiness and discriminate between "good" and "bad" debts, depending on how likely applicants are to default with their repayments (Lim and Sohn 2007). Compared with the subjective methods, automatic credit scoring models present a number of interesting advantages (Rosenberg and Gleit 1994, Thomas et al. 2002, Blochlinger and Leippold 2006): (i) reduction in the cost of the credit evaluation process and the expected risk of being a bad loan; (ii) time and effort savings; (iii) consistent recommendations based on objective information, thus eliminating human biases and prejudices; (iv) facilities to incorporate changes in policy and/or economy into the system; and (v) the performance of the credit scoring model can be monitored,

tracked, and adjusted at any time.

From the seminal reference to credit scoring in the introductory paper by Altman (1968), many other developments have been subsequently proposed (Baensens et al. 2003, Bahrammirzaee 2010, Abdou and Pointon 2011). After the Basel II recommendations issued by the Basel Committee on Banking Supervision in 2004, it has become almost a regulatory requirement for banks and financial organizations to utilize advanced credit scoring models in order for enhancing the efficiency of capital allocation. The effects of Basel II on the competitive advantage of an institution are so large that research on the adequacy, applicability and validity of the adopted systems is set to be a cardinal research initiative (Crook et al. 2007).

The most classical approaches to credit scoring employ statistical methods (discriminant analysis, linear and logistic regression, multivariate adaptive regression splines, classification and regression trees, nonparametric smoothing, survival analysis) or operations research models (linear programming, quadratic programming, integer programming, multiple criteria programming, dynamic programming), but it is also possible to find more sophisticated techniques belonging to the area of computational intelligence (often referred to as data mining or soft computing) such as neural networks, support vector machines, fuzzy systems, rough sets, artificial immune systems, and evolutionary algorithms.

The evolutionary computing methods are highly capable of extracting meaning from imprecise data and detecting trends that are too complex to be discovered by either humans or other conventional techniques. However, Fensterstock (2005) points out two main weaknesses for the application of these techniques to credit risk assessment: (i) they require a major computational effort, and (ii) they continue to be seldom known by the financial and business analysts. Despite these criticisms, one can find a considerable number of articles where evolutionary computing has been applied to credit scoring, what reflects the interest of the scientific community in this subject.

The aim of this review article is to synthesize, analyze and evaluate relevant

evolutionary computing techniques (considered as a subfield of computational intelligence) that have been applied to credit risk management, exploring the most recent investigation trends and identifying weaknesses and suggestions for further research. The review will focus on the three major areas or problem types where evolutionary computing has been used in credit scoring: classification, variable selection, and parameter optimization.

Hereafter, the paper is organized as follows. Section 2 gives a brief introduction to the evolutionary computing techniques. Section 3 outlines the research methodology, defines credit scoring as a classification problem, and provides a general description of feature or variable selection. Section 4 examines the existing research on evolutionary computation applied to credit scoring. Section 5 discusses the key points of the review, putting the emphasis on the shortcomings and the potential problems and topics that should be addressed in the future. Finally, Section 6 summarizes the general findings from this study.

2 Evolutionary computation

Computational intelligence is defined as the study of adaptive mechanisms to enable or facilitate intelligent behavior in complex and changing environments (Bezdek 1994, Engelbrecht 2007). These mechanisms include artificial intelligence concepts, paradigms, algorithms and implementations that exhibit an ability to learn or adapt to new situations, to generalize, abstract, discover and associate.

To tackle complex real-world problems, scientists have been looking into natural processes and creatures, both as model and metaphor. Optimization is at the heart of many natural processes including Darwinian evolution, social group behavior and foraging strategies. Over the last few decades, there has been growing interest in the field of biologically or nature-inspired search and optimization algorithms. Currently these techniques are applied to a variety of problems, ranging from scientific research to finance, industry and commerce to name but a few. The two main families of methods that primarily comprise this field are evolutionary

computing and swarm intelligence.

In particular, evolutionary computation constitutes a subfield of computational intelligence, which involves combinatorial optimization problems. Thus evolutionary computing is the generic term for a set of problem-solving techniques based on the Darwinian principles of natural selection and evolution (Eiben and Smith 2007). It is inspired by biological processes of inheritance, mutation, natural selection, and the genetic crossover that occurs when parents mate to produce offspring. It makes use of the concept of survival of the fittest by progressively accepting better solutions to the problem.

The common underlying idea behind all variants of evolutionary algorithms is that, given a population of individuals, the environment causes natural selection and this produces a rise in the fitness of the population. Given a quality function to be maximized, we can randomly generate a set of candidate solutions (a group of IF-THEN rules) and apply the quality function as an abstract fitness measure. Based on this, some of the better candidates are chosen to seed the next generation by applying selection, mutation and crossover operators. These operators lead to a set of new candidates that compete with the old ones for a place in the next generation. This process can be iterated until a candidate with sufficient quality is found or a previously fixed computational limit is reached.

In general, evolutionary algorithms are stochastic search algorithms that operate on a population of candidate solutions. Traditionally, they refer to four categories of similar but independently developed techniques (Fogel 2000): genetic algorithms, evolutionary strategies, evolutionary programming, and genetic programming. From these, the genetic algorithms and genetic programming approaches have been the most widely used in credit scoring applications.

2.1 Genetic algorithms and genetic programming

Genetic algorithms (Holland 1975, 1992) provide a method to perform randomized global search in a solution space. They operate on a population of potential solutions applying the principle of survival of the fittest to produce (hopefully)

better and better approximations to a solution. At each generation, a new set of approximations is created by the process of selecting individuals according to their level of fitness in the problem domain and breeding them together using operators borrowed from natural genetics. This process leads to the evolution of populations of individuals that are better suited to their environment than the individuals that they were created from.

Usually, the algorithm starts with a random population of N candidate solutions, which are internally encoded as chromosomes (in the form of a string). Next the quality of each chromosome x in the population is evaluated by a fitness function $\phi(x)$, and the best two are selected to crossover and form a new solution (offspring). A further genetic operator, called mutation, may be then applied to the new offspring, which causes the individual genetic representation to be changed according to some probabilistic rule. After recombination and mutation, the process continues through subsequent generations and it terminates either after a predefined number of iterations or if the best member of the latest populations has not improved during a certain number of iterations.

The three most important aspects of using genetic algorithms are: (i) definition of the fitness function, (ii) definition and implementation of the genetic representation, and (iii) definition and implementation of the genetic operators.

On the other hand, genetic programming proposed by Koza (1992) is an extension to the original concept of genetic algorithms. The population in genetic programming is composed by variable length tree-like candidate solutions whose size, shape, and complexity can dynamically change during the process. Each of these individual candidates, called program, may have functional nodes, enabling the solution to perform arbitrarily large actions. Genetic programming also uses crossover and mutation as the transformation operators to change candidate solutions into new candidate solutions as genetic algorithms do.

Differences between genetic programming and genetic algorithms refer to the particular representation of the solution: genetic programming produces computer programs or programming language expressions as the solution, whereas genetic

algorithms give a string of numbers that represent the solution. The major advantage of genetic algorithms and genetic programming is that one does not have to specify all the details of a problem in advance, but potential solutions are evaluated by a fitness function representing the problem to be solved.

3 Research methodology

In order to review the literature that deals with evolutionary computation applied to credit scoring, two groups of keywords were used to cross-search for related papers in seven databases of scientific publications: ProQuest ABI/Inform, Google Scholar, IEEE Xplore, Emerald, ScienceDirect, ACM Digital Library, and Scopus. The first group of descriptors includes computational intelligence, soft computing, artificial intelligence, data mining, machine learning, evolutionary computation, evolutionary programming, genetic algorithm, and genetic programming; the second group comprises credit scoring, credit risk, financial risk, credit evaluation, credit assessment, credit analysis, credit decision, and credit management.

The present review was confined to conference papers and journal articles published within the period 2000–2012 in order to explore and analyze the current state-of-the-art of credit scoring models based on evolutionary computation techniques. The reason why this review focuses on works from the last decade is that there already exist some reviews of past literature (up to 2000) relative to Statistics and Artificial Intelligence applied to general topics of finance, accounting, human resources, marketing, production, and distribution (Dimitras et al. 1996, Hand and Henley 1997, Refenes et al. 1997, Wong et al. 1997, Vellido et al. 1999, Thomas 2000, Metaxiotis and Psarras 2004), but to the best of our knowledge not specifically to credit scoring.

The methodological framework of this research was defined by analyzing a variety of problem types related to general applications of computational intelligence. Thus the articles were classified into three main groups according to the purpose with which the evolutionary computing techniques have been applied to

credit scoring: (i) classification, (ii) variable selection, and (iii) parameter optimization. In addition, a fourth group was also included to gather a miscellany of other problems barely covered by evolutionary methods.

3.1 Classification

Classification is one of the main applications of intelligent systems (Komar 2005) and from a practical point of view, credit scoring can be deemed as a classical classification or prediction problem where a new input sample (customer) must be categorized into one of the predefined classes based on a number of observed variables (features, attributes) related to that sample. In this context, the input of the classifier consists of a variety of information that describes socio-demographic characteristics and economic conditions of the applicant (age, gender, number of dependents, loan purpose, annual income, marital status, educational level, occupation, number of years employed, house owned or rented, estimated assets), and the credit scoring model will produce the output in terms of the customer creditworthiness. In its most usual form, credit scoring aims at assigning credit applicants to one of two classes: good customers (those who are liable to reimburse the financial obligation) and bad customers (those who should be denied credit because of the high probability of defaulting on repayments).

More formally, given a data set of n customers, $S = \{(x_1, y_1), \dots, (x_n, y_n)\}$, where each customer $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ is characterized by D variables defined on an input feature space X^D , and $y_i \in \{good, bad\}$ denoting the type of customer, then a credit scoring model (the classifier) can be defined as a mapping function $f : X^D \rightarrow \{good, bad\}$ that predicts the value y for a new credit applicant \mathbf{x} , that is, $f(\mathbf{x}) = y$.

There are five major characteristics of intelligent systems that make them especially appealing for solving financial and banking problems, such as is the case of credit scoring and assessment (Goonatilake and Treleaven 1995):

- Learning: The ability to learn decisions and tasks from historical data and/or

previous examples.

- **Adaptation:** The ability to adapt to a changing environment such as policy changes, new government regulations or changes in economic conditions.
- **Flexibility:** The ability to make decisions and perform tasks even in the presence of incomplete or unreliable data.
- **Transparency:** The ability to explain how the decisions were reached. In some cases, due to legal or organizational reasons, it is compulsory to provide explanations of how decisions have been made.
- **Discovery:** The ability to discover new processes or relationships that were previously unknown.

3.2 Variable selection

Variable or feature selection is the process of finding the best subset from a given set of variables in a dataset (Dash and Liu 1997). This is an important issue in designing classification systems such as scorecards. It is imperative to select the most relevant variables and to limit the number of input features in a scorecard in order to have a more predictive and less computationally intensive model. Furthermore, with a smaller set of variables to decide upon credit approval, the scorecard analyst will gain more insight into the model and better comprehensibility of the decisions made.

The objective of variable selection is three-fold (Guyon and Elisseeff 2003): improving the performance of the predictors, providing faster and more cost-effective predictors, and providing a better understanding of the underlying process that generated the data.

Suppose a large set of D features is given from which we have to find a small subset d according to some optimality criteria. Although it may not be apparent, the subset selection problem may become prohibitive because of its computational complexity. In general, it will not be possible to rank the variables simply based

on their individual properties and select only the best. The feature properties may depend strongly on each other and a subset of individually weak variables may prove to be rather good because of positive interaction effects. Because of this uncertainty, the only apparent way of searching for optimal subsets is to evaluate all the 2^D possible variable combinations. However, testing all subsets is a combinatorial problem that requires an exponential amount of computational time and consequently, a broad range of sub-optimal feature selection methods has been developed over the last decades.

Traditionally, two families of variable selection methods have been considered: filters and wrappers (George 2000). Filter techniques use some evaluation function (information theoretic measures, distance measures, consistency measures, correlation measures) that relies solely on properties of the data, thus being independent of the choice of the learning algorithm used to construct the classifier; they attempt to remove irrelevant and redundant variables before training the model. Conversely, wrapper methods utilize the learning algorithm itself to evaluate the candidate subsets of features (Kohavi and John 1997). In its most general formulation, the wrapper methodology consists of using the prediction performance of a given learning algorithm to assess the relative usefulness of all possible feature subsets. It is argued that, compared to wrappers, filters propose a faster and generic selection of variables, not tuned for a particular learning algorithm (Guyon and Elisseeff 2003).

4 Evolutionary computing in credit scoring

This section surveys relevant research works carried out in the context of evolutionary computation applied to credit scoring. The review has been organized around the main problems tackled through the development of genetic algorithms and genetic programming. Table 1 provides a summary with the articles that will be analyzed in the next sections.

Table 1: Summary of research in the period 2000–2012

Problem type	Reference	Benchmark methods
Classification	Fritz and Hosemann (2000)	LDA, ANN, KNN, M6
	Yobas et al. (2000)	LDA, ANN, C4.5
	Hoffmann et al. (2002, 2007)	Fuzzy rules, ANN, C4.5, Bayes, LDA
	Mukerjee et al. (2002)	—
	Chen and Huang (2003)	Ensembles
	Bedingfield and Smith (2003)	—
	Ong et al. (2005)	ANN, CART, C4.5, LogR, rough sets
	Tsakonas et al. (2006)	ANN, C4.5, fuzzy rules
	Huang et al. (2006)	ANN, CART, C4.5, KNN, LogR, rough sets
	Huang et al. (2007)	ANN, SVM, C4.5
	Zhang et al. (2007)	Ensembles, ANN, SVM
	Finlay (2006, 2009)	ANN, LogR, OLS regression
	Cai et al. (2009)	—
	Zhang et al. (2008a)	Ensembles, SVM, C4.5, LogR, ANN
	Abdou (2009)	LogR, WOE
Jiang et al. (2011)	ANN	
Variable selection	Eklund et al. (2001)	Ad hoc model
	Sexton et al. (2006), Sustersic et al. (2009), Oreski et al. (2012)	ANN
	Huang et al. (2007)	SVM
	Liu et al. (2008)	LDA
	Marinakos et al. (2008)	KNN
	Zhang et al. (2008b)	Ensembles, C4.5, SVM, ANN
	Wang and Huang (2009)	C4.5, ANN, KNN, SVM, naive Bayes, LDA
	Mahmoud et al. (2010)	C4.5
	Chi and Hsu (2012)	LogR, ANN
	Huang and Wu (2011)	Ensembles, LogR, KNN, ANN, C4.5, decision table, SVM
Parameter optimization	Huang et al. (2007), Zhou and Bai (2008), Zhou et al. (2009), Yu et al. (2011), Danenas and Garsva (2012), Li et al. (2012)	SVM
	Lahtasna et al. (2008), Aïnon et al. (2009)	SVM, CART, C4.5, ANN, fuzzy rules, rough sets
	Wang et al. (2008), de Pinho et al. (2009), Eletter et al. (2010), Fu and Liu (2011)	ANN
	Lacerda et al. (2005)	ANN, SVM, C4.5
	Yao (2009)	CART, SVM, MARS
	Mahmoud et al. (2010)	C4.5
	Xu et al. (2010)	Ensembles
	Li et al. (2011)	LDA, QDA, LogR, KNN, SVM, C4.5
	Correa and González (2011)	LogR, ANN
	Vukovic et al. (2012)	KNN
	Model selection	Finlay (2010)
Comprehensibility	Martens et al. (2007), Li et al. (2011)	—
Missing values	Sakprasat and Sinclair (2007)	—
Computing time	Hens and Tiwari (2012)	SVM
Instance selection	Tsai and Chou (2011), García et al. (2012)	SVM, KNN

4.1 Classification

In the context of the evolutionary approaches to credit scoring, one has a number of scorecards that mutate and blend together according to their fitness function at classification (Fogerty and Ireson 1993). One of the earliest papers comparing genetic algorithms with other credit scoring models is that by Yobas et al. (2000). This work compares the predictive performance of a traditional statistical method (linear discriminant analysis, LDA) with that of three computational intelligence techniques (a neural network, a decision tree and a genetic algorithm), using a small sample of credit scoring data. The authors find that linear discriminant analysis is superior to genetic algorithms and neural networks. Fritz and Hosemann (2000) also conclude that the linear discriminant analysis model is better than genetic algorithms and three other approaches to credit scoring, including artificial neural networks, the M6 decision tree and the k -nearest neighbors decision rule (KNN). However, validity of their conclusions can be questioned because they do not utilize the same training and test sets for experimenting with the different techniques. In fact, these results are inconsistent with those of other previous studies (Desai et al. 1996, 1997, Varetto 1998). More recently, Ong et al. (2005) observe that credit classification based on standard genetic programming outperforms other models constructed with artificial neural networks (ANN), decision trees (CART and C4.5), rough sets, and logistic regression (LogR).

Huang et al. (2007) compare the performance of genetic programming against neural network, support vector machine (SVM) and C4.5 decision tree models in a credit scoring application using the Australian and German benchmarking databases. Their study reveal better classification accuracy with genetic programming than with the other techniques, although differences are marginal.

Tsakonas et al. (2006) evaluate four computational intelligence methods with a sample of enterprises that applied for a loan in a European banking organization, paying special attention to the comprehensibility and the ease of use for the acquired decision models. Two methods employ grammar-guided genetic programming for the generation of either classification trees (Tsakonas and Dounias

2002) or fuzzy rule-based systems (Tsakonas et al. 2001). The approaches based on genetic programming appears to perform rather low, but comprehensibility and generalization ability are higher.

Mukerjee et al. (2002) present a novel tool for the application of a multi-objective genetic algorithm, namely NSGA-II (Deb 2001), to a bank credit management problem, with the aim of giving an appropriate trade-off between the multiple objectives of return maximization and risk minimization. The new technique produces an approximation to the set of Pareto-optimal solutions, which may increase the decision flexibility available to the bank manager, and is also computationally efficient when compared to the traditional multi-objective method of epsilon-constraints.

Finlay (2006, 2009) introduces an evolutionary approach to the implementation of credit scoring models, demonstrating that genetic algorithms can perform as well as, if not marginally better than, competing models constructed using neural networks, OLS regression and logistic regression. For the experiments, data are supplied by Experian UK and contain details of credit applications made in a time period.

Chen and Huang (2003) combine genetic algorithms and artificial neural networks for inverse classification in order to reassign the rejected instances (bad customers) to the preferable accepted class (good customers), trying also to analyze the reasons for rejection of applicants. Zhang et al. (2007) develop a combined model based on back-propagation neural networks, genetic programming and support vector machines, concluding that it can perform better than any single classifier over the Australian and German databases. Jiang et al. (2011) use simulated annealing to optimize a genetic algorithm and construct a combined credit scoring model, which results with the highest prediction rates in comparison to back-propagation and RBF neural networks.

Cai et al. (2009) present a genetic algorithm that employs an appropriate fitness evaluation function, taking into account the values of both good and bad customers in the training set; experiments over the German credit database reflect

an acceptable accuracy rate on the class of good customers, but poor performance on the samples belonging to bad customers.

The problem of classifying credit data with unequal misclassification costs is studied by Bedingfield and Smith (2003): an evolutionary algorithm is employed to generate systems of classification rules. In addition to the misclassification costs, various other properties of the classification systems generated by the evolutionary algorithm, such as accuracy and coverage, are also considered and discussed.

A two-stage genetic programming (called 2SGP) is introduced by Huang et al. (2006) to address the credit scoring problem by incorporating the advantages of the IF-THEN rules and the discriminant functions. From the experimental results over the German and Australian credit databases, the authors conclude that 2SGP is more flexible and performs significantly better than the other models tested (multi-layer perceptron neural network, CART and C4.5 decision trees, k -nearest neighbors rule, rough sets and logistic regression).

Zhang et al. (2008a) propose a hybrid credit scoring model by profiting from the advantages of genetic programming and support vector machines. In a first stage, genetic programming is used to derive simple decision rules and in a second stage, those samples that have not satisfied any rules or have satisfied more than one rule are employed to train a support vector machine and build the discriminant function. In this paper, genetic programming utilizes a reduced set of logical connectives (AND, OR, NOT), the relation operators (\leq , $=$, \geq), and the conditional operator (IF-THEN). Moreover, the variables with continuous values are first discretized because many studies have shown that rules with discrete values are often shorter and more understandable and on the other hand, discretization can lead to improved predictive accuracy. The hybrid model is empirically compared with support vector machines, single genetic programming, C4.5 decision tree, logistic regression, and back-propagation neural network, demonstrating better accuracy of the hybrid approach over the German and Australian databases.

Abdou (2009) investigates the ability of genetic programming in the analysis

of credit scoring models using data from Egyptian public sector banks, comparing this technique with probit analysis (a well-studied alternative to logistic regression (Dionne et al. 1996)) and weight of evidence (WOE) measure (a credit scoring model that focuses on the odds ratio of good scores to bad scores (Bailey 2001)). Experimental results show that genetic programming achieves the highest accuracy rate and also the lowest type-I and type-II errors when compared to the other two techniques. However, the genetic programming approach does not provide the lowest estimated misclassification cost, which corresponds to WOE.

Despite the powerful learning capabilities of genetic algorithms, one major criticism of applying them to credit scoring is their poor comprehensibility (Goldberg 1989). Several solutions to this opacity problem consist of combining genetic algorithms with other techniques that can be better understood by humans. For instance, Hoffmann et al. (2002) employ a boosted genetic fuzzy classification rule (Cordón and Herrera 1998) on real credit scoring data, finding that it performs better than the NefClass neuro-fuzzy algorithm and the C4.5 decision tree. Similarly, descriptive fuzzy rules are extracted from a genetic algorithm, where all fuzzy rules share a common, linguistically interpretable definition of membership functions (Hoffmann et al. 2007). In this work, the genetic fuzzy rules outperform other classifiers in terms of classification accuracy when applied to the German and Australian databases and two credit scoring data sets from Benelux financial institutions.

4.2 Variable selection

An important application field of evolutionary computing is feature selection. For example, Eklund et al. (2001) utilize genetic programming to choose the most explanatory variables for credit risk analysis. Liu et al. (2008) argue that in credit scoring applications, characteristics derived from the input data variables are more discriminative and may indicate cross-information among those attributes. Because the process to select this kind of characteristics can be viewed as a combinatorial optimization problem of mathematical symbols and original attributes,

the authors propose a hybrid method with genetic programming and human interaction to determine the derived variables with most practical significance. These characteristics are then used in a linear discriminant analysis credit scoring model, concluding that this may reduce the risk of credit prediction.

Sexton et al. (2006) apply a genetic-based algorithm, called the neural network simultaneous optimization algorithm, to a credit approval data set. The algorithm determines those input variables that contribute to estimation of the underlying function. This can help with analysis of the problem, improve generalization, and reducing the size of the neural network structure for credit scoring.

In the paper by Huang et al. (2007), genetic algorithms are used to simultaneously perform feature selection and parameter optimization in a support vector machine. For the Australian data set, the support vector machine using genetic algorithms achieves better accuracy than the other two approaches using grid search and F -score, and even with a smaller number of variables selected; the results over the German database are similar in terms of accuracy and also with regards to the number of features selected.

Sustersic et al. (2009) develop a back-propagation neural network credit scoring model with a data preprocessing stage to select a pseudo-optimal subset of the input features. To this end, the authors apply a genetic-based feature selection algorithm (Zupan and Novic 1999), producing an important reduction in the number of variables and yet with high classification accuracy. Their initial hypothesis is that the variables with poor information content and co-linear variables might introduce noise to the model, thus leading to a decrease in accuracy.

With the idea of using evolutionary computation to preprocess the data, the K -means clustering algorithm and a genetic algorithm are combined to further improve the accuracy of a credit scoring model based on a C4.5 decision tree (Zhang et al. 2008b). The genetic algorithm is firstly utilized to decrease the number of attributes (by removing redundant variables) and make the decision tree simpler and more understandable, while the clustering algorithm aims at removing noisy data. Computational results show that the use of genetic algorithms and K -means

clustering can effectively improve the prediction rate of the credit scoring models analyzed, including a decision tree, a neural network, a support vector machine and a random subspace classifier.

Marinakakis et al. (2008) propose to optimize the nearest neighbor decision rule through three metaheuristic algorithms: tabu search (Glover 1989, 1990), genetic algorithms and ant colony optimization (Dorigo and Stutzle 2004). The three algorithms are tested over data derived from the loan portfolio of the Commercial Bank of Greece in a credit risk assessment problem. The paper concludes that such algorithms can be successfully applied to the feature selection problem, leading to models that provide good classification results with a small number of attributes. In particular, the ant colony optimization approach appears to be the method with the best accuracy using almost half of the original features.

Similarly, Wang and Huang (2009) use the so-called non-dominated sorting genetic algorithm (Deb et al. 2000) with inter-correlation and intra-correlation for feature selection. The basic idea is to find out the best attribute subset with the property of lower intra-correlation within the set of variables, but higher inter-correlation between the set and the corresponding class, thereby making the set discriminatory. The authors apply the new feature selection algorithm to the Australian and German databases using various classifiers: back-propagation and perceptron multi-layer neural networks, C4.5 decision tree, nearest neighbor rule, support vector machine, naive Bayes classifier, and linear discriminant analysis.

Chi and Hsu (2012) select relevant variables by means of a genetic algorithm to construct a dual scoring model for credit risk management of mortgage accounts, which combines the bank internal behavioral scoring model with the external credit bureau scoring model. It undergoes more accurate risk judgment and segmentation to further discover the parts that are required to be enhanced in management or control from mortgage portfolio.

The study by Huang and Wu (2011) compares several prediction models taken from data mining methods, and improves traditional credit scoring techniques by using boosting and genetic algorithms for feature selection. The empirical results

indicate that the use of genetic algorithms improves the performance of underlying classifiers and appears to be more robust than the single prediction methods.

Oreski et al. (2012) investigate the performance of seven feature selection techniques over a data set from a Croatian bank, showing that the genetic algorithm gives superior results compared to the other approaches. However, when considering different misclassification costs, the evolutionary-based feature selection method appears to be worst than some other techniques. The authors point out that this is due to the fact that the genetic algorithm intends to optimize the accuracy rate, without taking into account other particular conditions.

4.3 Parameter optimization

Since genetic algorithms are basically an optimization heuristic, Lacerda et al. (2005) investigate the use of RBF neural networks designed by means of genetic algorithms and concluded that such an implementation is superior (lowest average classification error and smallest average number of hidden nodes) to other methods in a credit assessment application. Besides, while the other techniques have their parameters defined by a trial-and-error iterative process, the genetic-based approach automatically searches for the parameters of the RBF neural network. A similar but more recent approach (Zhou and Bai 2008, Zhou et al. 2009) utilizes a genetic algorithm to search for proper parameters of a support vector machine and compares it with other three optimization methods, showing no significant differences in the experimental accuracies. Yu et al. (2011) investigate the performance of genetic algorithms for parameter selection in a weighted least squares support vector machine and find that other optimization methods outperform the evolutionary approach.

Wang et al. (2008) establish a hybrid neural network based on the combination of genetic and back-propagation algorithms and apply the model in personal credit scoring of commercial banks. The genetic algorithm is applied to optimize the initial weights and biases of a neural network. The experimental results indicate that such a hybrid model can improve the learning ability of a neural network

effectively and overcome the drawbacks of the back-propagation algorithm. Analogously, Fu and Liu (2011) construct a hybrid personal credit scoring model by combining RBF neural networks with a genetic algorithm; the new model employs the genetic algorithm to optimize weights and other parameters of the RBF neural network, so that convergence of the model is rapid and presents good generalization over the German database.

With the aim of obtaining both accuracy and interpretability, an evolutionary-neuro-fuzzy method is adopted in the works by Lahsasna et al. (2008) and Aïnon et al. (2009). In a first phase, a fuzzy rule base is automatically extracted from a data set using a clustering method, and then a genetic algorithm is used to increase the performance of the fuzzy inference system. Finally, a multi-objective genetic algorithm is applied to preserve the accuracy of the fuzzy model to a predefined value and enhance the interpretability of the fuzzy model by reducing the fuzzy sets in the rule base.

A quantum-inspired genetic algorithm for a credit assessment application using a neural approach is proposed by de Pinho et al. (2009). Specifically, the genetic algorithm is employed to completely configure a feed-forward neural network in terms of selecting the relevant input variables, the number of neurons in the hidden layer and all synaptic weights. The experiments over the Australian database prove the effectiveness of the new model in comparison to other techniques, providing good classification accuracy and high degree of flexibility.

Yao (2009) defines three strategies to construct hybrid fuzzy support vector machine credit scoring models: using the CART decision tree to select input variables, using multivariate adaptive regression splines (MARS) to select input features, and using a genetic algorithm to optimize model parameters. The experiments carried out over the Australian and German databases demonstrate that the hybrid model has the best overall classification accuracy and also the lowest type-II error.

Eletter et al. (2010) support the use of genetic algorithms to improve the process of finding better parameters of a multi-layer feed-forward neural network

model for a credit scoring application in the Jordanian commercial banks. Also, Li et al. (2011) propose a multiple kernels multi-criteria programming model, which optimizes the parameters based on the idea of natural evolution. Experiments over the Australian and German databases achieve successful results in terms of both the classification accuracy and the number of features selected, when compared to a number of prediction models that include linear discriminant analysis, quadratic discriminant analysis (QDA), support vector machine, C4.5 decision tree, logistic regression and k -nearest neighbors rule.

Mahmoud et al. (2010) claim that a genetic algorithm can help to build optimal C4.5 decision trees by selecting appropriate features and generating better decision trees for credit scoring. Under this assumption, the model proposed by the authors selects and combines the best decision tree based on a given optimality criterion and constructs the final decision tree for credit scoring of customers.

Xu et al. (2010) propose a novel support vector machine based ensemble model for credit risk assessment. Firstly, the method employs principle component analysis for feature selection, and then some support vector machines with different kernels are trained by using a genetic algorithm to optimize the parameters. Finally, all results produced by different support vector machines are combined by several ensemble strategies.

It is accepted that artificial neural networks are powerful tools for classification and regression, but it is difficult and time costly to determine the best architecture for a given problem. Correa and González (2011) use genetic algorithms to optimize the architecture of a multi-layer perceptron, in order to improve the predictive power of the credit risk scorecards. The objective function to maximize is the ROC curve and the input variables are the number of hidden layers and units, activation function, use or not of bias and whether it will be a direct connection between the initial and the final layer. Results show that this method outperforms logistic regression and the default neural network architecture.

Vukovic et al. (2012) employ genetic algorithms to set the parameters of each preference function and the weights of the attributes in a case-based reasoning

model that uses preference theory functions for measuring similarity between samples. More specifically, the model corresponds to the k -nearest neighbors classifier, and the preference functions are based on the Promethee multi-criteria decision-making method. Although the experiments over three credit databases indicate that the k -nearest neighbors rule optimized by genetic algorithms may outperform the traditional model, the study presents some limitations that should be taken into account for further research.

In the paper by Li et al. (2012), the parameters of an adaptive support vector machine with a Gaussian kernel are optimized through an evolutionary strategy. The advantages of using a genetic algorithm instead of grid search to conduct parameter optimization refer to very important savings in computation time, what is demonstrated by experiments over the Australian and German benchmarking databases and a real-life US commercial bank credit data set. Danenas and Garsva (2012) employ genetic algorithms and particle swarm optimization to select the parameters of a support vector machine.

4.4 Other problems covered by evolutionary techniques

As a result of the opacity property of the support vector machines, they can sometimes become difficult to apply to credit risk evaluation. To overcome this limitation, rules can be extracted from the trained machine that are easily interpretable by humans and keep as much of the accuracy of the support vector machine as possible. Martens et al. (2007) introduce a rule extraction technique based on genetic programming, which is then tested over two credit scoring databases: the Australian benchmarking data set and one obtained from a major Benelux financial institution. The experiments show that the support vector machine rule extraction technique using genetic programming lose only a small percentage in performance compared to support vector machines, whereas it ranks at the top of comprehensible classification models.

Another problem common to many financial applications is the presence of missing data (samples with missing values in some variables), what may intro-

duce bias and result in misleading conclusions drawn from the analysis carried out. Sakprasat and Sinclair (2007) focus on how to handle missing data in a credit approval database; the experimental results show that genetic programming can be successfully used with credit data that contain missing values, although it appears to be preferable to preprocess the data and then apply the genetic approach for classification.

Finlay (2010) compares predictive models of continuous financial behavior with binary (good/bad payer) models of customer default, concluding that the use of continuous models for consumer risk assessment outperforms the traditional binary classification approaches. This work also suggests that scoring functions developed to specifically optimize the profit contribution, using genetic algorithms, outperform scoring functions derived from optimizing more general functions such as the sum of squared errors.

Hens and Tiwari (2012) study the computational time required by two implementations of support vector machines (one is based on genetic algorithms, whereas the other uses stratified sampling), an artificial neural network and genetic programming. The authors conclude that the neural network and the evolutionary models can become too costly because they need to optimize a number of parameters in order for achieving accuracy rates similar to those of the support vector machine based on stratified sampling.

García et al. (2012) explore the behavior of twenty noise filtering techniques applied to instance-based credit risk prediction models. The experiments over eight credit databases demonstrate that the evolutionary algorithms obtain the highest set size reduction rates, but no significant differences are achieved in terms of accuracy. On the other hand, Tsai and Chou (2011) employ a genetic algorithm to simultaneously perform feature selection and data reduction as a preprocessing stage when modeling a support vector machine and k -nearest neighbors rule, concluding that instance selection is much more important than feature selection in the domain of credit risk prediction.

5 Discussion and challenges

From the literature review conducted in the previous section, it can be observed that the application of evolutionary computing techniques to credit scoring is an emerging trend that has attracted much attention of practitioners and academics in the last decade. As can be seen in Table 1, this paradigm of computational intelligence has been used for credit risk prediction, but mostly for feature selection and/or parameter optimization as a preprocessing stage that allows to increase the performance of other classification models.

While credit scoring systems based on evolutionary computation have been tried with some success in terms of predictive power, they have often been criticized because they are considered as black-box techniques whose resulting decisions are not easily interpretable for the financial and business analysts (Lucas 2001). This is one of the main reasons put forward by decision makers to utilize other more comprehensible classifiers, such as rule-based models and decision trees, rather than evolutionary techniques. In other cases, the genetic algorithms and genetic programming have been combined with other models aiming at both accuracy and interpretability (Tsakonas et al. 2006, Hoffmann et al. 2007, Lahsasna et al. 2008, Zhang et al. 2008a,b).

Also, it is worth noting that, in a large percentage of the articles here analyzed, the evolutionary algorithms have solely been included to be compared with other models and techniques (see, for example, the studies by Yobas et al. (2000), Eklund et al. (2001), Ong et al. (2005), Huang et al. (2007), Marinakis et al. (2008), Abdou (2009), Cai et al. (2009), Yu et al. (2011), Chi and Hsu (2012), Oreski et al. (2012), García et al. (2012), but there are just a few new developments or modifications of existing algorithms. Examples of relatively new implementations are those proposed by Mukerjee et al. (2002), Bedingfield and Smith (2003), Huang et al. (2006), Zhang et al. (2008a), de Pinho et al. (2009).

One of the key points for the evolutionary algorithms to provide good performance is to set up properly several internal parameters, such as the fitness function, the population size, the maximum number of generations, the mutation rate

and the crossover rate. For example, if the mutation rate is too high, the system will never converge towards a stable solution; conversely, if it is too low, the population will usually converge on some local optimum. Albeit this is an important issue that should be clearly addressed in any scientific publication, we have perceived that only a few papers indicate whether or not they have estimated the optimal values of the parameters. Some authors have not even reported the specific parameter settings of the models used in their experiments. The problem is that, without this information, it is difficult for another researcher to replicate the experimentation with the aim of providing a point of comparison for new developments.

The review carried out also raises other relevant issues that will be addressed in the next sections. In particular, a discussion on the performance evaluation measures, the statistical significance tests and the databases used in the experiments may be of great interest to the reader in order for identifying any deficiencies and gaps in existing publications that should be further explored in future research on the application of evolutionary computation to credit scoring. Finally, we also discuss some issues on data preprocessing and modeling that have not been tackled through evolutionary computation.

5.1 Performance evaluation criteria

A frequent problem encountered in the literature reviewed refers to the variety of criteria employed for assessing the model performance. What is even more important is that some of these metrics, however, are not the most appropriate to be used in credit scoring applications because of the different misclassification costs associated to each class (good customers and bad customers).

The accuracy or hit rate, which measures the proportion of the correctly predicted cases, has been considered as a significant criterion in evaluating the classification capability of the scoring models and in fact, it corresponds to the performance evaluator most commonly used by the authors (more than 60% of the papers here studied). However, empirical and theoretical evidences have demon-

strated that the accuracy rate is strongly biased with respect to imbalance in class distribution and ignores the different misclassification costs (Fawcett and Provost 1997). In real applications, it is generally believed that the class of bad customers is under-represented in comparison to the class of good customers and in addition, the costs associated with type-II errors (bad customers misclassified as good) are much higher than the misclassification costs associated with type-I errors (good customers mispredicted as bad) (West 2000, Baesens et al. 2003).

Even with the important drawbacks related to the accuracy rate, less than 20% of the papers report the type-I and type-II errors, and only two articles (Abdou 2009, Oreski et al. 2012) employ the estimated misclassification cost (the relative costs of accepting credit applications that become bad versus rejecting applications that would be good). The problem with the estimated misclassification cost criterion is that it is really difficult to obtain reliable estimates of costs.

The receiver operating characteristic (ROC) curve is one of the most widely used methods for visualization and comparison of the performance of classifiers. It is a two-dimensional graph that plots the proportion of bad cases predicted as bad (sensitivity or true positive rate) versus the proportion of good cases predicted as bad (one minus specificity) at all cut-off score values. In the ROC space, the upper left corner corresponds to perfect classification, whereas a diagonal line represents random classification. The ROC curve shows the behavior of models independently of misclassification costs and class distributions (Thomas et al. 2002). Despite its usefulness, only three papers of this review (Wang and Huang 2009, Correa and González 2011, Chi and Hsu 2012) have employed ROC analysis to assess the performance of the models.

Other metrics used in a few papers include the mean squared error, the Gini coefficient, the area under the ROC curve, the return rate, the Kolmogorov-Smirnov statistic, the sensitivity and specificity (or true negative rate), the coefficient of concordance, the F -measure, and the precision.

Finally, it should be noted that only a small percentage of the studies utilize some measure of efficiency, such as CPU time (Chen and Huang 2003, Sexton

et al. 2006, Liu et al. 2008, Zhou et al. 2009, Correa and González 2011, Hens and Tiwari 2012), number of rules generated (Hoffmann et al. 2002, 2007, Martens et al. 2007, Lahsasna et al. 2008, Aïnon et al. 2009) and decision tree size (Mahmoud et al. 2010). These two latter metrics, the number of rules and the decision tree size, can also be viewed as a way to measure the comprehensibility or transparency of the model, which has become a very important factor for realistic credit scoring systems.

One can conclude that there does not exist an optimal performance evaluation criterion, but it depends on a number of factors relative to each particular application. More specifically, the choice of a measure or another seems to be determined both by the problem in hand and by the availability of certain data (e.g., the misclassification costs).

5.2 Statistical significance tests

It is worth noting that most of the articles here reviewed do not contain any formal statistical test, so it is unclear whether or not the apparent differences in performance are significant.

On the other hand, some papers (Lacerda et al. 2005, Hoffmann et al. 2007, Martens et al. 2007, de Pinho et al. 2009, Oreski et al. 2012) include the paired t-test for assessing statistical significance of difference, but this appears to be conceptually inappropriate and statistically unsafe because parametric tests assume independence, normality and homogeneity of variance, which are often violated due to the nature of the problems (Demšar 2006).

In general, the non-parametric procedures, such as the Wilcoxon signed ranks test and the Friedman test with the corresponding post-hoc tests, should be preferred over the parametric ones. In the present review, Sexton et al. (2006), García et al. (2012) and Vukovic et al. (2012) report a Wilcoxon signed rank test to statistically compare the performance of the algorithms, whereas Finlay (2010) employs a Kruskal-Wallis test to show that differences between pairs of techniques are significant.

5.3 Credit databases

Many authors have experimented only with small benchmarking databases such as the well-known German and Australian databases (Frank and Asuncion 2010) and therefore, the conclusions drawn from a limited amount of results should be taken cautiously because these data sets might not be representative enough of real-life credit scoring applications.

There are also other works that have used data gathered from different sources, such as a bankruptcy data set from a Benelux financial institution (Martens et al. 2007), a data set provided by a subsidiary of the Construction Bank of China (Zhou and Bai 2008), a loan portfolio of the Commercial Bank of Greece (Marinakis et al. 2008), personal loan data sets from banks in Egypt (Abdou 2009), data supplied by Experian UK (Finlay 2006, 2009), a consumer loan database from Jordanian banks (Eletter et al. 2010), data from a US commercial bank (Li et al. 2012), or a credit data set provided by a Croatian bank (Oreski et al. 2012). These papers mainly focuses on a single database, converting their models into ad-hoc solutions that can become hard to use under other different conditions.

Unfortunately, there are relatively few studies that have conducted experiments over a sufficiently large number of data sets, thus allowing to generalize their conclusions (see, for example, the investigation by García et al. (2012)). This issue is especially important in the case of evolutionary algorithms because they require to set up a lot of parameters whose optimal values strongly depend on the characteristics of each database, among other things.

Although it is not easy to access to customer credit data from banks and financial institutions due to the laws and regulations on security of data and privacy protection of customer information, it seems necessary to create a public repository of credit data that can become available to the academic community, with certain confidentiality restrictions to prevent the identification of individual customers.

5.4 Some data preprocessing issues

Data preprocessing constitutes a critical step in developing high performance classification and prediction systems. The primary aim is to derive a subset of data, which is expected to be a representative sample of the problem to be modeled (Pyle 1999, Dasu and Johnson 2003).

There are different tools and methods used for data preprocessing, each one focused on solving some specific problem related to data quality (Zhang et al. 2005). Thus the purpose of sampling is to minimize the size of very large data sets, whereas filtering or cleaning aims at removing noise from data; discretization converts continuous attributes into discrete data, and normalization is a scaling-down transformation of the attributes; feature selection and extraction pull out as much irrelevant and redundant information as possible. Other outstanding issues in real-life credit scoring applications include reject inference (process whereby the performance of previously rejected applicants is inferred primarily using the repayment behavior of accepted applicants), class imbalance (usually referred to as the low-default portfolio problem in the context of credit scoring), missing values in attributes, and population drift (changes occurring in the population due to unexpected changes in economic conditions and other factors).

The credit data sets are usually heavily imbalanced, that is, the number of default cases is typically very low, what may cause the estimate of probability of default to be statistically unreliable (Flórez-López 2010, Kennedy et al. 2010). Also, typically we have only information about the repayment behavior of those who were granted for credit in the past, but not of those previous applicants who were rejected; a model built using data only on accepted applicants will be biased when employed for all applicants. Reject inference investigates on how to include those previous applicants in the modeling process in order to make the classifier more accurate and less biased (Crook and Banasik 2004). This situation can become even worse with the existence of incomplete data, which may be caused by many circumstances (for example, loss of information or refusal of applicants to answer certain questions) (Flórez-López 2010). On the other hand,

if changes occur in the population of applicants, the credit risk prediction model needs some adaptive learning strategy in order to avoid a significant deterioration in performance (Pavlidis et al. 2012).

While feature selection has widely been handled by means of evolutionary computation, the other problems have not been addressed in depth. As already seen in Section 4.4, Sakprasat and Sinclair (2007) utilize genetic programming to deal with the presence of missing data in a credit approval database, Tsai and Chou (2011) conduct feature and instance selection with a genetic algorithm to model a support vector machine and a k -nearest neighbors rule, and García et al. (2012) use several evolutionary algorithms to remove noisy samples and reduce the size of the data set. However, no more articles related to data preprocessing with evolutionary computing in credit scoring have been found in the literature review.

6 Conclusions

The future of credit risk analysis is an increased reliance on computerized credit scoring models. Automatic decision-making will never take the place of the financial analyst, but it can help make quick, consistent decisions to approve or deny the majority of transactions that fall above or below certain credit score thresholds.

This paper has presented a literature review on recent applications of evolutionary techniques to credit scoring and assessment. Now it is generally admitted that genetic algorithms and genetic programming are efficient, flexible methods to find optimal or near-optimal solutions from the search space, what makes them especially appealing to a wide variety of economic and financial applications. In this sense, we have found that evolutionary computation has mainly been applied to variable selection and parameter optimization as two important preprocessing steps for further classification with other prediction models, especially artificial neural networks and support vector machines.

An important subject in credit scoring is the comprehensibility or transparency

of the model, which has become a key factor for the lending industry. In general, it has been seen that evolutionary algorithms should preferably be combined with (neuro)-fuzzy rules or decision trees to achieve a higher degree of transparency. Another promising method in this direction is the so-called multi-objective genetic algorithm because it is suitable for systems with conflicting goals, which in the case of credit scoring are to increase the predictive power and to reduce the complexity of the models. On the other hand, the use of evolutionary computation in variable selection leads to a reduction of the complexity, what also allows to increase the comprehensibility of the model.

When focusing on classification, one can observe that there is no single best algorithm across different credit databases. A technique may be the best on some particular data sets, but it will perform worse than the other algorithms on other different data sets. Therefore it is not possible to conclude that genetic algorithms and genetic programming are better or worse than other models, but are simply an alternative to conventional methods. However, the major interest of using evolutionary methods probably comes from their ability to properly optimize the internal parameters of those classifiers that have demonstrated to perform well in credit scoring applications.

The present review has also revealed some weaknesses and limitations that should be taken into account in the future: (i) the lack of credit databases publicly available makes difficult to perform extensive comparisons between methods and thus, several papers have merely included experiments over the German and Australian benchmarking data sets; (ii) there exists a great variety of performance evaluation criteria that have been used in the articles, but some are not adequate due to the different misclassification costs and/or imbalanced class distributions often encountered in credit scoring problems; (iii) the conclusions drawn by most authors have not been supported by statistical tests to demonstrate the significance of the experimental results and, in other cases the statistic applied is inappropriate because of the incorrect assumptions they have made; (iv) although data pre-processing is a very important task in classifier modeling and involves a lot of

problems or imperfections in data (e.g., missing data, population drift, noisy examples, low-default portfolio), it has been observed that variable selection is the only problem that has largely been tackled with evolutionary techniques.

Finally, it is also interesting to point out that most research in evolutionary computation applied to credit scoring has been published by journals that belong to the field of Computer Science, especially *Expert Systems with Applications*, but also others such as *Applied Soft Computing*, *Decision Support Systems*, *Soft Computing*, *International Journal of Intelligent Systems*, and *Applied Intelligence*. A second group of journals with articles on this topic can be deemed as interdisciplinary: *European Journal of Operational Research*, *Journal of Forecasting*, *Operations Research*, and *Journal of Global Optimization*. Paradoxically, despite the need of cooperation between the scientific communities of Economics and Computer Science, only a limited number of papers on evolutionary computing applications to credit scoring has been published by journals that belong to fields such as Business, Economics, Finance, or Management (for example, *Journal of Banking & Finance*, *American Journal of Economics and Business Administration*, *Economic Bulletin*, and *Intelligent Systems in Accounting, Finance & Management*).

Acknowledgment

This work has partially been supported by the Spanish Ministry of Education and Science under grant TIN2009–14205 and the Generalitat Valenciana under grant PROMETEO/2010/028.

References

- Abdou, H. A. 2009. Genetic programming for credit scoring: The case of Egyptian public sector banks. *Expert Systems with Applications* **36**(9) 11402–11417.
- Abdou, H. A., J. Pointon. 2011. Credit scoring, statistical techniques and evaluation

- criteria: A review of the literature. *Intelligent Systems in Accounting, Finance and Management* **18**(2–3) 59–88.
- Abrahams, C. R., M. Zhang. 2008. *Fair Lending Compliance: Intelligence and Implications for Credit Risk Management*. Wiley, Hoboken, NJ.
- Ainon, R. N., A. Lahsasna, T. Y. Wah. 2009. A transparent classification model using a hybrid soft computing method. *Proc. 3rd Asia International Conference on Modelling and Simulation*. Bali, Indonesia, 146–151.
- Altman, E. I. 1968. Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *Journal of Finance* **23**(4) 589–611.
- Baesens, B., T. Van Gestel, S. Viaene, M. Stepanova, J. Suykens, J. Vanthienen. 2003. Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society* **54**(6) 627–635.
- Bahrammirzaee, A. 2010. A comparative survey of artificial intelligence applications in finance: artificial neural networks, expert system and hybrid intelligent systems. *Neural Computing and Applications* **19**(8) 1165–1195.
- Bailey, M. 2001. *Credit Scoring: The Principles and Practicalities*. White Box Publishing, Bristol, UK.
- Bedingfield, S., K. Smith. 2003. Predicting bad credit risk: An evolutionary approach. *Proc. Joint International Conference on Artificial Neural Networks and Neural Information Processing*. Istanbul, Turkey, 179–179.
- Bezdek, J. C. 1994. What is computational intelligence? J. M. Zurada, R. J. Marks II, C. J. Robinson, eds., *Computational Intelligence: Imitating Life*, chap. 1. IEEE Press, Piscataway, NJ, 1–12.
- Blochlinger, A., M. Leippold. 2006. Economic benefit of powerful credit scoring. *Journal of Banking & Finance* **30**(3) 851–873.
- Cai, W., X. Lin, X. Huang, H. Zhong. 2009. A genetic algorithm model for personal credit scoring. *Proc. International Conference on Computational Intelligence and Software Engineering*. Wuhan, China, 1–4.
- Chen, M.-C., S.-H. Huang. 2003. Credit scoring and rejected instances reassigning through evolutionary computation techniques. *Expert Systems with Applications* **24**(4) 433–441.

- Chi, B.-W., C.-C. Hsu. 2012. A hybrid approach to integrate genetic algorithm into dual scoring model in enhancing the performance of credit scoring model. *Expert Systems with Applications* **39**(3) 2650–2661.
- Cordón, O., F. Herrera. 1998. Genetic learning of fuzzy rule-based classification systems cooperating with fuzzy reasoning methods. *International Journal of Intelligent Systems* **13**(10–11) 1025–1053.
- Correa, A., A. González. 2011. Evolutionary algorithms for selecting the architecture of a MLP neural network: A credit scoring case. *Proc. 11th IEEE International Conference on Data Mining Workshops*. Vancouver, Canada, 725–732.
- Crook, J., J. Banasik. 2004. Does reject inference really improve the performance of application scoring models? *Journal of Banking & Finance* **28**(4) 857–874.
- Crook, J. N., D. B. Edelman, L. C. Thomas. 2007. Recent developments in consumer credit risk assessment. *European Journal of Operational Research* **183**(3) 1447–1465.
- Danenas, P., G. Garsva. 2012. Credit risk evaluation modeling using evolutionary linear SVM classifiers and sliding window approach. *Procedia Computer Science* **9** 1324–1333.
- Dash, M., H. Liu. 1997. Feature selection for classification. *Intelligent Data Analysis* **1**(1–4) 131–156.
- Dasu, T., T. Johnson. 2003. *Exploratory Data Mining and Data Cleaning*. John Wiley & Sons, Hoboken, NJ.
- de Pinho, A. G., M. Vellasco, A. V. A. da Cruz. 2009. A new model for credit approval problems: A quantum-inspired neuro-evolutionary algorithm with binary-real representation. *Proc. World Congress on Nature & Biologically Inspired Computing*. Coimbatore, India, 445–450.
- Deb, K. 2001. *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley, Chichester, UK.
- Deb, K., S. Agrawal, A. Pratap, T. Meyarivan. 2000. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimisation: NSGA-II. *Proc. 6th International Conference on Parallel Problem Solving from Nature*. Paris, France, 849–858.

- Demšar, J. 2006. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research* **7**(1) 1–30.
- Desai, V. S., D. G. Conway, J. N. Crook, G. A. Overstreet. 1997. Credit-scoring models in the credit-union environment using neural networks and genetic algorithms. *IMA Journal of Management Mathematics* **8**(4) 323–346.
- Desai, V. S., J. N. Crook, G. A. Overstreet. 1996. A comparison of neural networks and linear scoring models in the credit union environment. *European Journal of Operational Research* **95**(1) 24–37.
- Dimitras, A. I., S. H. Zanakis, C. Zopounidis. 1996. A survey of business failures with an emphasis on prediction methods and industrial applications. *European Journal of Operational Research* **90**(3) 487–513.
- Dionne, G., M. Artís, M. Guillen. 1996. Count data models for a credit scoring system. *Journal of Empirical Finance* **3**(3) 303–325.
- Dorigo, M., T. Stutzle. 2004. *Ant Colony Optimization*. MIT Press, Cambridge, MA.
- Eiben, A. E., J. E. Smith. 2007. *Introduction to Evolutionary Computing*. Springer, Heidelberg, Germany.
- Eklund, T., K. Larsen, E. Berghardsen. 2001. Model for analysing credit risk in the enterprise sector. *Economic Bulletin* **72**(3) 99–106.
- Eletter, S. F., S. G. Yaseen, G. A. Elrefae. 2010. Neuro-based artificial intelligence model for loan decisions. *American Journal of Economics and Business Administration* **2**(1) 27–34.
- Engelbrecht, A. P. 2007. *Computational Intelligence: An Introduction*. Wiley, Chichester, UK.
- Fawcett, T., F. Provost. 1997. Adaptive fraud detection. *Data Mining and Knowledge Discovery* **1**(3) 291–316.
- Fensterstock, A. 2005. Credit scoring and the next step. *Business Credit* **107**(3) 46–49.
- Finlay, S. M. 2006. Using genetic algorithms to develop scoring models for alternative measures of performance. Tech. Rep. 2006/022, Department of Management Science, Lancaster University, UK.
- Finlay, S. M. 2009. Are we modelling the right thing? The impact of incorrect problem specification in credit scoring. *Expert Systems with Applications* **36**(5) 9065–9071.

- Finlay, S. M. 2010. Credit scoring for profitability objectives. *European Journal of Operational Research* **202**(2) 528–537.
- Flórez-López, R. 2010. Effects of missing data in credit risk scoring. a comparative analysis of methods to achieve robustness in the absence of sufficient data. *Journal of the Operational Research Society* **61**(3) 486–501.
- Fogel, D. B. 2000. What is evolutionary computation. *IEEE Spectrum* **72**(2) 26–32.
- Fogerty, T. C., N. S. Ireson. 1993. Evolving Bayesian classifiers for credit control—a comparison with other machine learning methods. *IMA Journal of Management Mathematics* **5**(1) 63–75.
- Frank, A., A. Asuncion. 2010. UCI Machine Learning Repository. Tech. rep., University of California, Irvine, School of Information and Computer Sciences. URL <http://archive.ics.uci.edu/ml>.
- Fritz, S., D. Hosemann. 2000. Restructuring the credit process: Behaviour scoring for German corporates. *Intelligent Systems in Accounting, Finance & Management* **9**(1) 9–21.
- Fu, H., X. Liu. 2011. A hybrid model for credit evaluation problem. *Proc. 2nd International Conference on Advances in Swarm intelligence*. Chongqing, China, 626–634.
- García, V., A. I. Marqués, J. S. Sánchez. 2012. On the use of data filtering techniques for credit risk prediction with instance-based models. *Expert Systems with Applications* (to appear).
- George, E. I. 2000. The variable selection problem. *Journal of the American Statistical Association* **95**(452) 1304–1308.
- Glover, F. 1989. Tabu search—Part I. *Inform Journal on Computing* **1**(3) 190–206.
- Glover, F. 1990. Tabu search—Part II. *Inform Journal on Computing* **2**(1) 4–32.
- Goldberg, D. E. 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Boston, MA.
- Goonatilake, S., P. Treleaven. 1995. *Intelligent Systems for Finance and Business*. Wiley, New York.
- Guyon, I., A. Elisseeff. 2003. An introduction to variable and feature selection. *Journal of Machine Learning Research* **3** 1157–1182.

- Hand, D. J., W. E. Henley. 1997. Statistical classification methods in consumer credit scoring: A review. *Journal of the Royal Statistical Society: Series A* **160**(3) 523–541.
- Hens, A. B., M. K. Tiwari. 2012. Computational time reduction for credit scoring: An integrated approach based on support vector machine and stratified sampling method. *Expert Systems with Applications* **39**(8) 6774–6781.
- Hoffmann, F., B. Baesens, J. Martens, F. Put, J. Vanthienen. 2002. Comparing a genetic fuzzy and a neurofuzzy classifier for credit scoring. *International Journal of Intelligent Systems* **17**(11) 1067–1083.
- Hoffmann, F., B. Baesens, C. Mues, T. Van Gestel, J. Vanthienen. 2007. Inferring descriptive and approximate fuzzy rules for credit scoring using evolutionary algorithms. *European Journal of Operational Research* **177**(1) 540–555.
- Holland, J. H. 1975. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, MI.
- Holland, J. H. 1992. *Adaptation in Natural and Artificial Systems*. MIT Press, Cambridge, MA.
- Huang, C.-L., M.-C. Chen, C.-J. Wang. 2007. Credit scoring with a data mining approach based on support vector machines. *Expert Systems with Applications* **33**(4) 847–856.
- Huang, J.-J., G.-H. Tzeng, C.-S. Ong. 2006. Two-stage genetic programming (2SGP) for the credit scoring model. *Applied Mathematics and Computation* **174**(2) 1039–1053.
- Huang, S.-C., C.-F. Wu. 2011. Customer credit quality assessments using data mining methods for banking industries. *African Journal of Business Management* **5**(11) 4438–4445.
- Jiang, M., F. Ji, R. Li. 2011. The applied research of credit scoring combination model based on SA-GA algorithm. *Proc. 4th International Conference on Business Intelligence and Financial Engineering*. Wuhan, China, 491–494.
- Kennedy, K., B. Mac Namee, S. J. Delany. 2010. Learning without default: a study of one-class classification and the low-default portfolio problem. *Proc. 20th Irish Conference on Artificial Intelligence and Cognitive Science*. Dublin, Ireland, 174–187.

- Kohavi, R., G. H. John. 1997. Wrappers for feature subset selection. *Artificial Intelligence* **97**(1–2) 273–324.
- Komar, A. 2005. *Computational Intelligence: Principles, Techniques and Applications*. Springer, Heidelberg, Germany.
- Koza, J. R. 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA.
- Lacerda, E., A. C. Carvalho, A. P. Braga, T. B. Ludermir. 2005. Evolutionary radial basis functions for credit assessment. *Applied Intelligence* **22**(3) 167–181.
- Lahsasna, A., R. N. Ainon, T. Y. Wah. 2008. Intelligent credit scoring model using soft computing approach. *Proc. International Conference on Computer and Communication Engineering*. Kuala Lumpur, Malaysia, 396–402.
- Li, J., G. Li, D. Sun, C.-F. Lee. 2012. Evolution strategy based adaptive Lq penalty support vector machines with Gauss kernel for credit risk analysis. *Applied Soft Computing* **12**(8) 2675–2682.
- Li, J., L. Wei, G. Li, W. Xu. 2011. An evolution strategy-based multiple kernels multi-criteria programming approach: The case of credit decision making. *Decision Support Systems* **51**(2) 292–298.
- Lim, M. K., S. Y. Sohn. 2007. Cluster-based dynamic scoring model. *Expert Systems with Applications* **32**(2) 427–431.
- Liu, S.-A., Q. Wang, S. Lv. 2008. Application of genetic programming in credit scoring. *Proc. Chinese Conference on Control and Decision*. Yantai, China, 1106–1110.
- Lucas, A. 2001. Statistical challenges in credit card issuing. *Applied Stochastic Models in Business and Industry* **17**(1) 83–92.
- Mahmoud, A., M. Pourzandi, K. Babaei. 2010. Using genetic algorithm in optimizing decision trees for credit scoring of banks customers. *Journal of Information Technology Management* **2**(4) 23–38.
- Marinakis, Y., M. Marinaki, M. Doumpos, N. Matsatsinis, C. Zopounidis. 2008. Optimization of nearest neighbor classifiers via metaheuristic algorithms for credit risk assessment. *Journal of Global Optimization* **42**(2) 279–293.
- Martens, D., B. Baesens, T. Van Gestel, J. Vanthienen. 2007. Comprehensible credit scor-

- ing models using rule extraction from support vector machines. *European Journal of Operational Research* **183**(3) 1466–1476.
- Metaxiotis, K., J. Psarras. 2004. The contribution of neural networks and genetic algorithms to business decision support: Academic myth or practical solution? *Management Decision* **42**(2) 229–242.
- Mukerjee, A., R. Biswas, K. Deb, A. Mathur. 2002. Multi-objective evolutionary algorithms for the risk-return trade-off in bank loan management. *International Transactions in Operational Research* **9**(5) 583–597.
- Ong, C.-S., J.-J. Huang, G.-H. Tzeng. 2005. Building credit scoring models using genetic programming. *Expert Systems with Applications* **29**(1) 41–47.
- Oreski, S., D. Oreski, G. Oreski. 2012. Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment. *Expert Systems with Applications* (to appear).
- Pavlidis, N. G., D. K. Tasoulis, N. M. Adams, D. J. Hand. 2012. Adaptive consumer credit classification. *Journal of the Operational Research Society* (to appear).
- Pyle, D. 1999. *Data Preparation for Data Mining*. Morgan Kaufmann Publishers, Los Altos, CA.
- Refenes, A.-P. N., A. Neil Burgess, Y. Bentz. 1997. Neural networks in financial engineering: A study in methodology. *IEEE Trans. on Neural Networks* **8**(6) 1222–1267.
- Rosenberg, E., A. Gleit. 1994. Quantitative methods in credit management: A survey. *Operations Research* **42**(4) 589–613.
- Sakprasat, S., M. C. Sinclair. 2007. Classification rule mining for automatic credit approval using genetic programming. *Proc. IEEE Congress on Evolutionary Computation*. Singapore, 548–555.
- Sexton, R. S., S. McMurtrey, D. Cleavenger. 2006. Knowledge discovery using a neural network simultaneous optimization algorithm on a real world classification problem. *European Journal of Operational Research* **168**(3) 1009–1018.
- Sustersic, M., D. Mramor, J. Zupan. 2009. Consumer credit scoring models with limited data. *Expert Systems with Applications* **36**(3) 4736–4744.
- Thomas, L. C. 2000. A survey of credit and behavioural scoring: Forecasting financial risk of lending to consumers. *International Journal of Forecasting* **16**(2) 149–172.

- Thomas, L. C., D. B. Edelman, J. N. Crook. 2002. *Credit Scoring and Its Applications*. SIAM, Philadelphia, PA.
- Tsai, C.-F., J.-S. Chou. 2011. Data pre-processing by genetic algorithms for bankruptcy prediction. *Proc. IEEE International Conference on Industrial Engineering and Engineering Management*. Singapore, 1780–1783.
- Tsakonas, A., N. Ampazis, G. Dounias. 2006. Towards a comprehensible and accurate credit management model: Application of four computational intelligence methodologies. *Proc. International Symposium on Evolving Fuzzy Systems*. Ambleside, UK, 295–299.
- Tsakonas, A., G. Dounias. 2002. Hierarchical classification trees using type-constrained genetic programming. *Proc. 1st International IEEE Symposium on Intelligent Systems*. Varna, Bulgaria, 50–54.
- Tsakonas, A., G. Dounias, H. Axer, D. G. V. Keyserlingk. 2001. Data classification using fuzzy rule-based systems represented as genetic programming type-constrained trees. *Proc. 8th UK Computational Intelligence Workshop*. Edinburgh, UK, 162–168.
- Varetto, F. 1998. Genetic algorithms applications in the analysis of insolvency risk. *Journal of Banking & Finance* **2**(10–11) 1421–1439.
- Vellido, A., P. J. G. Lisboa, J. Vaughan. 1999. Neural networks in business: A survey of applications (1992–1998). *Expert Systems with Applications* **17**(1) 51–70.
- Vukovic, S., B. Delibasic, A. Uzelac, M. Suknovic. 2012. A case-based reasoning model that uses preference theory functions for credit scoring. *Expert Systems with Applications* **39**(9) 8389–8395.
- Wang, C.-M., Y.-F. Huang. 2009. Evolutionary-based feature selection approaches with new criteria for data mining: A case study of credit approval data. *Expert Systems with Applications* **36**(3) 5900–5908.
- Wang, S., S. Yin, M. Jiang. 2008. Hybrid neural network based on GA-BP for personal credit scoring. *Proc. 4th International Conference on Natural Computation*. Washington, DC, USA, 209–214.
- West, D. 2000. Neural network credit scoring models. *Computers & Operations Research* **27**(11–12) 1131–1152.

- Wong, B.-K., T. A. Bodnovich, Y. Selvi. 1997. Neural networks applications in business: A review and analysis of the literature (1988–1995). *Decision Support Systems* **19**(4) 301–320.
- Xu, W., S. Zhou, D. Duan, Y. Chen. 2010. A support vector machine based method for credit risk assessment. *Proc. 7th International Conference on e-Business Engineering*. Shanghai, China, 50–55.
- Yao, P. 2009. Hybrid fuzzy SVM model using CART and MARS for credit scoring. *Proc. International Conference on Intelligent Human-Machine Systems and Cybernetics*, vol. 2. Hangzhou, China, 392–395.
- Yobas, M. B., J. N. Crook, P. Ross. 2000. Credit scoring using neural and evolutionary techniques. *IMA Journal of Mathematics Applied in Business and Industry* **11**(4) 111–125.
- Yu, L., X. Yao, S. Wang, K. K. Lai. 2011. Credit risk evaluation using a weighted least squares SVM classifier with design of experiment for parameter selection. *Expert Systems with Applications* **38**(12) 15392–15399.
- Zhang, C., Q. Yang, B. Liu. 2005. Guest Editors' Introduction: Special Section on Intelligent Data Preparation. *IEEE Trans. on Knowledge and Data Engineering* **17**(9) 1163–1165.
- Zhang, D., M. Hifi, Q. Chen, W. Ye. 2008a. A hybrid credit scoring model based on genetic programming and support vector machines. *Proc. 4th International Conference on Natural Computation*. Washington, USA, 8–12.
- Zhang, D., H. Huang, Q. Chen, Y. Jiang. 2007. A comparison study of credit scoring models. *Proc. 3rd International Conference on Natural Computation*. Haikou, China, 15–18.
- Zhang, D. F., S. Leung, Z. M. Ye. 2008b. A decision tree scoring model based on genetic algorithm and K-means algorithm. *Proc. 3rd International Conference on Convergence and Hybrid Information Technology*. Daejeon, Korea, 1043–1047.
- Zhou, J., T. Bai. 2008. Credit risk assessment using rough set theory and GA-based SVM. *Proc. 3rd International Conference on Grid and Pervasive Computing*. Washington, USA, 320–325.

- Zhou, L., K. K. Lai, L. Yu. 2009. Credit scoring using support vector machines with direct search for parameters selection. *Soft Computing* **13**(2) 149–155.
- Zupan, J., M. Novic. 1999. Optimisation of structure representation for QSAR studies. *Analytica Chimica Acta* **388**(3) 243–250.