Genetic algorithms for credit scoring: Alternative fitness function performance comparison

CrossMark

Vaclav Kozeny

Department of Economics and Development, Faculty of Tropical AgriSciences, Czech University of Life Sciences Prague, Czech Republic

A B S T R A C T

Credit scoring methods have been widely investigated by researchers; recently, genetic algorithms have attracted particular attention. Many research papers comparing the performance of genetic algorithms and traditional scoring techniques have been published, but most do not provide enough detail about the fitness function used by the genetic algorithm—despite the fact that fitness function has a key influence on the model's overall performance. The aim of this paper is to evaluate the predictive performance of different fitness functions used by genetic algorithms in credit scoring. An alternative fitness function based on a variable bitmask is proposed, and its performance then compared with fitness functions based on a polynomial equation as well as an estimation of parameter range. The results suggest that the bitmask is superior to the two other methods in both accuracy and sensitivity. The Wilcoxon matched-pairs sign rank test and paired t-Test indicate these results are statistically significant.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

Extending credit to the public is a core business of banks worldwide; the primary decision they face is whether to grant a loan to a potential customer. It is therefore essential that financial institutions are able to accurately differentiate between good and bad payers: this ability is limited by the data available to the bank at the time of application screening. Various credit-scoring methods have been developed to assist with this process. The most common ones, based on logistic regression, linear discriminant analysis, or k-Nearest Neighbor, are summarized by Vojtek and Kocenda (2006). In consumer lending, scoring methods draw largely on socio-demographic characteristics provided by clients in their loan application form. In their study, Avery, Calem, and Canner (2004) demonstrated that besides this rather static information, dynamic events in an individual's life can have a significant impact on their credit worthiness. Unfortunately, this type of information is hard to obtain. Individual default risk is also important from a regulatory perspective, as it contributes to the portfolio risk of the bank which is monitored by supervisory bodies. However, a simple addition of these risks may not be the best indicator of the total portfolio risk. In this context Jacobson and Roszbach (2003) proposed a method of weighting individual default risk estimates and applying them to the portfolio valuation model based on value-at-risk.

Generally in lending practice it is not sufficient to have a scorecard developed as it needs to be constantly validated as the market and demographic conditions change. Scorecard development and validation has been the focus of various studies (Dinh & Kleimeier, 2007; Lopez & Saidenberg, 2000; Wu & Olson, 2010). Furthermore, macroeconomic conditions usually influence the bank's overall lending policy as they have a global influence on market conditions (Bonfim, 2009; Stiglitz & Weiss, 1981).

Given the importance of credit scoring and its potential impact on a bank's business, it is unsurprising that traditional ways of assessing the credit worthiness of individuals are constantly being updated. Numerous studies comparing the performance of traditional and modern methods have been, and are being, conducted. For example, a comprehensive comparison of machine learning models with a traditional expert system was published by Ben-David and Frank (2009).

Biologically inspired techniques such as neural networks and genetic algorithms (GA) are becoming increasingly popular: their predictive power in credit scoring is being researched and compared with traditional models. Some studies indicate that these techniques can produce more accurate predictions (Desai, Conway, Crook, & Overstreet, 1997) than traditional approaches but other studies suggest they are less accurate (Fogarty & Ireson, 1993) or report mixed results (Desai, Crook, & Overstreet, 1996; Finlay, 2009). A review of the current state-of-the-art approaches to financial distress definition and prediction modeling was published by Sun, Li, Huang, and He (2014). A concise summary of the research conducted during the last decade in the field of evolutionary computing with its application to credit scoring has been published by Marques, Garcia, and Sanchez (2013).

Genetic algorithms were first introduced by Holland (1975) as an abstraction of biological evolution. A genetic algorithm uses genetic inspired operators to evolve an initial population into a new population. Each population comprises of chromosomes that represent genetically encoded individual solutions to a specific problem. Each individual has a fitness score assigned to them, which represents its ability in terms of a solution. A new population is evolved by using operators of crossover, mutation, and selection, where selection is based on the individual's fitness and influences its ability to reproduce into the next generation. Detailed information about different genetic operators, their functions and usage can be found in Mitchell (1998) or Michalewicz (1996).

The performance of genetic algorithms depends to a large degree on the parameters which are under the control of the researcher, requiring adjustments to deal with the specific problem at hand. These parameters and namely the fitness function therefore have to be carefully selected to match the specifics of credit scoring.

In current credit scoring research, GAs have been used in two different ways. The first area of application is a hybrid approach in which GAs are being used with other methods such as neural networks. In their research, Sustersic, Mramor, and Zupan (2009) use GAs to preselect the variables to be used by neural networks and logistic regression to develop a scoring model. Similarly, Chi and Hsu (2012) use GAs to preselect variables for their dual scoring model construction. This model comprises of both the credit bureau scoring model and the bank's own scoring model. Oreski, Oreski, and Oreski (2012) used a combination of GAs and neural networks to preselect variables and subsequently build a scoring model. Oreski and Oreski (2014) build on their previous research of GAs, and neural networks. They propose a method of incorporating feature selection into the GAs which provides a higher fitness starting population and faster convergence to optimum solution. Chen and Huang (2003) developed a scoring model using neural networks and then used GAs to provide more insight into the group of rejected applicants by conditional reclassification. An application of GAs to estimate validity constraints for the case-based reasoning model is presented by Vukovic, Delibasic, Uzelac, and Suknovic (2012).

The second area of application is the use of GAs as a complete standalone method. Gordini (2014) used genetic algorithms to generate classification rules for SME bankruptcy prediction. Competitive results have been achieved by Finlay (2009), who compared the performance of logistic and linear regression with GAs using a linear fitness function. Most literature, however, does not give enough detail as to the type of fitness function used. A description of a polynomial fitness function can be found in Thomas, Edelman, and Crook (2002). Another approach was proposed by Yobas, Crook, and Ross (2000), who used an estimation of parameter ranges as a fitness function.

These experiments were conducted using different data samples under different conditions. To the best of the author's knowledge, no study has been published comparing different approaches to fitness function selection using the same dataset.

It is the aim of this paper to propose an alternative fitness function based on a variable bitmask, investigate its performance, and compare it with the predictive ability of GAs using a polynomial fitness function, and with GAs using variable range estimation fitness function.

2. Materials and methods

Credit scoring can be described as a classification problem. Traditionally clients have been classified into two groups—*good*

and *bad*. This paper adopts the traditional approach but alternative approaches are also possible. A study classifying clients into three groups—good, poor and bad— has been published by Desai et al. (1997). Different studies propose methods to additionally reclassify the rejected groups (Chen & Huang, 2003; Chuang & Lin, 2009; Kim & Sohn, 2004). Some researchers claim that clients should be classified based on profit or net present value they bring to the bank. For example, Finlay (2010) uses GAs to construct profit maximizing scoring models, Blöchlinger and Leippold (2006) investigate ROC curves of scoring models with the aim of deriving a profit maximizing cut-off while Dionne, Artís, and Guillén (1996) extend traditional scoring model by inclusion of profit assessment based on collections costs.

The two-way classification problem can be described formally as follows:

Each customer x is classified by D variables $x = (x_1, x_2, \ldots, x_D)$, where each variable is of range V_j; $j \leqslant D$. The input feature space is then $V = \prod_{j=1}^{n} V_j = \{(x_1, \ldots, x_D) | x_j \in V_j\}$. A chromosome represents a mapping (scoring) function $f : V \rightarrow \{good, bad\}$ that predicts the type of a new credit applicant. The real observed client status in the sample is denoted as $y \in \{good, bad\}$. The fitness function is a combination of the mapping function and its corresponding fitness score. The training of the GA is performed on a client sample S with known characteristics and status:

$$S = \{(\vec{x}_1, y_1), \ldots, (\vec{x}_N, y_N)\} \tag{1}$$

where $\vec{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iD})$ is a client and y_i his corresponding status. The fitness score Φ is represented by accuracy calculated as the number of correct predictions divided by the total number of cases in sample N:

$$\Phi(f) = \#\{i \leqslant N | f(\vec{x}_i) = y_i\} / N \tag{2}$$

In this paper three main definitions of fitness function have been used: a fitness function defined by a polynomial equation, a fitness function using range estimates of each independent variable, and a fitness function based on a bitmask for every independent variable covering any combination of its possible values.

Since the paper focuses on fitness functions, to ensure comparability all models had key genetic operators set equally. Each of the genetic operators was fixed after experimentation with its alternatives. A final selection was made based on performance under the given technical constraints. Key characteristics were the ability to consistently reach higher optima solutions and the necessary time to do so.

Each model was initiated with the creation of an initial population of 200 chromosomes. The length of each chromosome in genes was dependent on the type of model as explained in Sections 2.1–2.3. The polynomial model had 23 genes, the range model had 30 genes and the bitmask model had 33 genes in each chromosome. After the model initiation a series of steps was carried out repeatedly. First, the fitness was calculated for every individual solution (chromosome) in the population. Subsequently all chromosomes were ranked based on their respective fitness scores and the elite 5% were copied unchanged to the next population. Additionally, forward migration was used copying 20% of the best fitness chromosomes to the next generation automatically every 20 generations. The third step was to select part of the population for crossover. Stochastic uniform sampling was applied as the selection method. This approach is similar to the popular roulette wheel selection method. The wheel can be constructed in various ways: one of the most frequent models used is fitness proportionate. In this case the wheel is divided into m sections where m equals the number of chromosomes in population. Each section then represents one chromosome; the size of the section is equal to its fitness. In this way, solutions with higher fitness have a greater