Available online at www.sciencedirect.com
ScienceDirect
European Journal of Operational Research 195 (2009) 942–959
EUROPEAN JOURNAL OF OPERATIONAL RESEARCH
www.elsevier.com/locate/ejor

An intelligent-agent-based fuzzy group decision making model for financial multicriteria decision support: The case of credit scoring

Lean Yu [a,b,*], Shouyang Wang [a], Kin Keung Lai [b]

[a] Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100080, China
[b] Department of Management Sciences, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong

Available online 12 November 2007

Abstract

Credit risk analysis is an active research area in financial risk management and credit scoring is one of the key analytical techniques in credit risk evaluation. In this study, a novel intelligent-agent-based fuzzy group decision making (GDM) model is proposed as an effective multicriteria decision analysis (MCDA) tool for credit risk evaluation. In this proposed model, some artificial intelligent techniques, which are used as intelligent agents, are first used to analyze and evaluate the risk levels of credit applicants over a set of pre-defined criteria. Then these evaluation results, generated by different intelligent agents, are fuzzified into some fuzzy opinions on credit risk level of applicants. Finally, these fuzzification opinions are aggregated into a group consensus and meantime the fuzzy aggregated consensus is defuzzified into a crisp aggregated value to support final decision for decision-makers of credit-granting institutions. For illustration and verification purposes, a simple numerical example and three real-world credit application approval datasets are presented.

Keywords: Multicriteria decision analysis; Fuzzy group decision making; Intelligent agent; Credit scoring; Artificial intelligence

1. Introduction

Without doubt credit risk evaluation is an important topic for research in the field of financial risk management. Generally, an accurate evaluation of credit risk could be transformed into a more efficient use of economic capital. When some customers fail to repay their debt, it leads to a direct economic loss for the lending financial organizations. If a credit-granting institution refuses loans to applicants with good credit scores, the institution loses the revenue it can earn from the applicant. On the other hand, if a credit-granting institution accepts applicants with bad credit scores, it may incur losses in the future – i.e. when the applicant fails to repay the debt. Therefore, credit risk evaluation is of extreme importance for lending organizations. Furthermore, credit risk evaluation has become a major focus of finance and banking industry due to the recent financial crises and regulatory concerns reflected in Basel II. For any credit-granting institution, such as a commercial bank or a retail financial company, the ability to discriminate good customers from bad ones is crucial for survival and development. The need for reliable models that can predict defaults accurately is imperative, in order to enable the interested parties to take either preventive or corrective action (Wang et al., 2005; Lai et al., 2006b,d).

In credit risk evaluation, credit scoring is one of the key analytical techniques. As Thomas (2002) defined, credit scoring is a technique that helps some organizations, such as commercial banks and credit card companies, determine whether or not to grant credit to consumers, on the basis of a set of predefined criteria. Usually, a credit score is a number that quantifies the creditworthiness of a person, based on a quantitative analysis of credit history and other criteria; it describes the

* Corresponding author. Address: Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100080, China. Tel.: +86 10 62565817; fax: +86 10 62568364.
E-mail address: yulean@amss.ac.cn (L. Yu).

0377-2217/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2007.11.025

extent to which the borrower is likely to pay his or her bills/debt. A credit score is primarily based on credit reports and information received from some major credit reporting agencies. Using credit scores, banks and credit card companies evaluate the potential risk involved in lending money, in order to minimize bad debts. Lenders can also use credit scores to determine who qualifies for what amount loan and at what interest rate. The generic approach of credit scoring is to apply a quantitative method on some data of previous customers – both faithful and delinquent customers – in order to find a relationship between the credit scores and a set of evaluation criteria. One important ingredient to accomplish this goal is to seek a good model so as to evaluate new applicants or existing customers as good or bad.

Due to the importance of credit risk evaluation, there is an increasing research stream focussing upon credit risk assessment and credit scoring. First of all, many statistical analysis and optimization methods, such as linear discriminant analysis (Fisher, 1936), logistic analysis (Wiginton, 1980), probit analysis (Grablowsky and Talley, 1981), linear programming (Glover, 1990), integer programming (Mangasarian, 1965), *k*-nearest neighbor (KNN) (Henley and Hand, 1996) and classification tree (Makowski, 1985), are widely applied to credit risk assessment and modeling tasks. Although these methods can be used to evaluate credit risk, the ability to discriminate good customers from bad ones is still a problem; the existing methods have their inherent limitations and can be improved further. Recent studies have revealed that emerging artificial intelligent (AI) techniques, such as artificial neural networks (ANNs) (Lai et al., 2006b,d; Malhotra and Malhotra, 2003; Smalz and Conrad, 1994), evolutionary computation (EC) and genetic algorithm (GA) (Chen and Huang, 2003; Varetto, 1998) and support vector machine (SVM) (Van Gestel et al., 2003; Huang et al., 2004; Lai et al., 2006a,c) are advantageous to statistical analysis and optimization models for credit risk evaluation in terms of their empirical results.

Although almost all classification methods can be used to evaluate credit risk, some combined or ensemble classifiers, which integrate two or more single classification methods, have turned out to be efficient strategies for achieving high performance, especially in fields where the development of a powerful single classifier system is difficult. Combined or ensemble modeling research is currently flourishing in credit risk evaluation. Recent examples are neural discriminant model (Lee et al., 2002), neuro-fuzzy model (Piramuthu, 1999; Malhotra and Malhotra, 2002), fuzzy SVM model (Wang et al., 2005) and neural network ensemble model (Lai et al., 2006b). A comprehensive review of literature about credit scoring and modeling is provided in two recent surveys (Thomas, 2002; Thomas et al., 2005).

Inspired by the combined or ensemble techniques, this study attempts to apply a group decision making (GDM) technique to support credit scoring decisions, using advanced computing techniques (ACTs). As is known to all, GDM is an active search field within multicriteria decision analysis (MCDA) (Beynon, 2005). In GDM, group members first make their own judgments on the same decision problems independently, i.e. decision actions, alternatives, projects and proposals and so on. These judgments from different group members are then aggregated to arrive at a final group decision. Different from the traditional GDM model, this study utilizes some artificial intelligence (AI) techniques to replace human experts. In the proposed approach, these AI agents can be seen as decision members of the decision group. Like human experts, these intelligent agents can also give some evaluation or judgment results on a specified problem, in terms of a set of predefined criteria. Relative to human experts' judgments, evaluation results provided by these intelligent agents (based on a set of criteria) are more objective because these intelligent agents are little affected by external considerations. Nevertheless, since some of the parameters and sampling of these intelligent agents are variable and unstable, these agents can often generate different judgments even though the same criteria are used. For handling these different judgments, we apply the fuzzification method. Thus the problem is further extended into a fuzzy GDM analytical framework. In this study, we try to propose an intelligent-agent-based fuzzy GDM model for credit scoring.

Generally, the proposed fuzzy GDM model is composed of three stages. In the first stage, some intelligent techniques as intelligent agents are used to analyze and evaluate the decision problems over a set of criteria. Because of different sampling and parameter settings, these intelligent agents may generate different judgments on the same decision problems. For handling these different judgments, the fuzzification method is utilized to formulate fuzzy judgments in the second stage. In the third stage, using classical optimization techniques and defuzzification method, these fuzzy opinions are finally aggregated into a group consensus as the final criterion for decision-making.

The purpose of this study is to propose an intelligent-agent-based fuzzy GDM model to support financial multicriteria decision making (MCDM) problems. Using the proposed model, many practical financial MCDM problems, such as enterprise financial condition diagnosis and financial risk analysis, can be solved effectively. For these real-world problems, decisions are made on the basis of a set of pre-defined criteria. Therefore, the proposed fuzzy GDM is suitable for solving these financial MCDM problems. As an illustration, a class of real-world MCDM problem concerning loan application approval is investigated in this study, using the credit scoring technique. Granting loan to applicants is an important financial decision problem, associated with credit risk of applicants, for most financial institutions. Usually, for applicants seeking small amounts of loans, the credit decision can be based on a standard scoring process. However, when amounts of loans are large, the decision-making process becomes more complex. In most situations, the decisions are made by a decision group not only because of the business opportunity at stake but also because of wider implications of the decision in terms of responsibility. DeSanctis and Gallupe (1987) highlight the reason for GDM – may be the problem is too significant for

any single individual to handle. In the customer loan application approval problem, most senior managers feel that opinions of other related members of the group, having some knowledge of the applicant, should be considered.

The main contribution of this study is that a fully novel intelligent-agent-based fuzzy GDM model is proposed for the first time, for solving a financial MCDM problem, by introducing some intelligent agents as decision-makers. Compared with traditional GDM methods, our proposed fuzzy GDM model has five distinct features. First of all, intelligent agents, instead of human experts, are used as decision-makers (DMs), thus reducing the recognition bias of human experts in GDM. Second, the judgment is made over a set of criteria through advanced intelligent techniques, based upon the data itself. Third, like human experts, these intelligent agents can also generate different possible opinions on a specified decision problem, by suitable sampling and parameter setting. All possible opinions then become the basis for formulating fuzzy opinions for further decision-making actions. In this way, the specified decision problems are extended into a fuzzy GDM framework. Fourth, different from previous subjective methods and traditional time-consuming iterative procedures, this article proposes a fast optimization technique to integrate the fuzzy opinions and to make the aggregation of fuzzy opinions simple. Finally, the main advantage of the fuzzy aggregation process in the proposed methodology is that it can not only speed up the computational process via information fuzzification but also keep the useful information as possible by means of some specified fuzzification ways.

The rest of this paper is organized as follows. In Section 2, the proposed intelligent-agent-based fuzzy GDM methodology is described in detail. For illustration and verification purposes, Section 3 presents a simple numerical example to illustrate the implementation process of the proposed fuzzy GDM model; three real-world credit datasets are used to test the effectiveness of the proposed fuzzy GDM model. In Section 4, some concluding remarks are drawn.

2. Methodology formulation

To illustrate the intelligent-agent-based fuzzy GDM model proposed in this paper, a practical financial MCDM problem – credit risk evaluation problem – is presented. As previously mentioned, granting credit to applicants is an important business decision problem for credit-granting institutions like commercial banks and credit card companies and credit scoring is one of the important techniques used in credit risk evaluation problems. In credit scoring, a generic process consists of two procedures: (1) applying a quantitative technique on similar data of previous customers – both faithful and delinquent customers – to uncover a relationship between the credit scores and a set of criteria; (2) utilizing the discovered relationship and new applicants' credit data to score new applicants and to evaluate new applicants as good or bad applicants.

From the above two procedures, it is not hard to find that machine learning and artificial intelligence (AI) techniques are very suitable for solving credit scoring problems. In machine learning and AI techniques, in-sample training and out-of-sample testing are the two required processes. In these two processes, the first corresponds to in-sample training and learning, while the second corresponds to out-of-sample testing and generalization.

As noted earlier, in case of large amounts of loan, the decision is usually determined by a group of decision-makers, over a set of criteria, thereby making the credit application approval become a GDM problem. The basic idea of the GDM model is to make full use of knowledge and intelligence of the members of a group to make a rational decision over a pre-defined set of criteria. Different from traditional GDM, the group members in this case are some artificial intelligent agents, instead of human experts. Suppose that there are n decision-makers (DM) as AI agents and m criteria for some decision problems or projects. Then the typical intelligent-agent-based multicriteria GDM model can be further illustrated as in Fig. 1.

For a specified decision problem or decision project, different decision-makers usually give different estimations or judgments over a set of criteria $X = (c_1, c_2, \ldots, c_m)$. For example, for a credit scoring problem, the decision makers may give the highest score (optimistic estimation), the lowest score (pessimistic estimation) and the most likely score, using a set of criteria and credit information of the applicants. In order to incorporate these different judgments of the decision-makers into the final decision and to make full use of the different judgments, a process of fuzzification is used. In the above example, a typical triangular fuzzy number can be used to describe the judgments of the decision-makers, i.e.

Problems/	DM_1 (AI agent)			DM_2 (AI agent)			.....			DM_n (AI agent)		
Projects	c_1	...	c_m	c_1	...	c_m	c_1	...	c_m	c_1	...	c_m
1												
...												
N												

Fig. 1. An illustrative sketch of the intelligent-agent-based multicriteria GDM model.

$$\tilde{Z}_i = (z_{i1}, z_{i2}, z_{i3}) = (\text{the lowest score}, \text{the most likely score}, \text{the highest score}), \tag{1}$$

where i represents the numerical index of decision-makers.

Like human experts, individual AI agents can also generate different judgment results by using different parameter settings and training sets. For example, for a credit scoring problem, the neural network agent generates k different judgments (i.e. k different credit scores) by setting different hidden neurons or different initial weights. That is, using a set of evaluation criteria X, the AI agent's output $Y = f(X)$ can be used as the applicant's credit score where the function $f(\cdot)$ is determined by the intelligent learning process. Note that our study mainly uses the final output value $f(X)$ of intelligent-agent-based models as the applicants' credit scores. Usually we use the following classification functions $F(X)$ to evaluate applicants as good or bad: $F(X) = \text{sign}(f(X) - T_\theta)$, where $f(X)$ is the output value of the three intelligent agents and T_θ is the credit threshold or cutoff. For a credit scoring problem, a credit analyst can adjust or modify the cutoff to change the percent of accepted applications. Only when an applicant's credit score is higher than the cutoff T_θ, his/her application will be accepted.

Assume that the ith decision-maker (DM_i, AI agent here) produces k different credit scores, $f_1^i(X_A), f_2^i(X_A), \ldots, f_k^i(X_A)$, for a specified applicant "A" over a set of criteria X. In order to make full use of all information provided by credit scores, without loss of generalization, we still utilize the triangular fuzzy number to construct the fuzzy opinion for consistency. That is, the smallest, average and the largest of the k credit scores are used as the left-, medium- and right-membership degrees. That is, the smallest and the largest scores are seen as optimistic and pessimistic evaluations and the average score is considered to be the most likely score. Of course, other fuzzified approaches to determining membership degree can also be used. For example, we can use the median as the most likely score to construct the triangular fuzzy number. But this way we may lose some useful information because some other scores are ignored. Therefore, we select the average as the most likely score to incorporate full information in all the scores into the fuzzy judgment. Using this fuzzification method, the decision-makers (DMs) can make a fuzzy judgment for each applicant. More precisely, the triangular fuzzy number for judgment, DM_i in this case, can be represented as

$$\tilde{Z}_i = (z_{i1}, z_{i2}, z_{i3}) = \left(\left[\min\left(f_1^i(X_A), f_2^i(X_A), \ldots, f_k^i(X_A)\right)\right], \left[\sum\nolimits_{j=1}^{k} f_j^i(X_A)/k\right], \left[\max\left(f_1^i(X_A), f_2^i(X_A), \ldots, f_k^i(X_A)\right)\right] \right). \tag{2}$$

In such a fuzzification process, the credit scoring problem is extended into a fuzzy GDM framework. Suppose that there are p DMs; let $\tilde{Z} = \psi(\tilde{Z}_1, \tilde{Z}_2, \ldots, \tilde{Z}_p)$ be the aggregation of the p fuzzy judgments, where $\psi(\cdot)$ is an aggregation function. Now how to determine the aggregation function or how to aggregate these fuzzy judgments into a group consensus is an important and critical problem under the GDM environment. Generally speaking, there are many aggregation techniques and rules that can be used to aggregate fuzzy judgments. Some of them are linear and others are non-linear. Interested readers may kindly refer to Cholewa (1985), Ramakrishnan and Rao (1992), Yager (1993, 1994), Delgado et al. (1998), Irion (1998), Park and Kim (1996), Lee (2002), Zhang and Lu (2003) and Xu (2004, 2005) for more details. Usually, the fuzzy judgments of the p group members will be aggregated by using a commonly used linear additive procedure, i.e.

$$\tilde{Z} = \sum_{i=1}^{p} w_i \tilde{Z}_i = \left(\sum_{i=1}^{p} w_i z_{i1}, \sum_{i=1}^{p} w_i z_{i2}, \sum_{i=1}^{p} w_i z_{i3} \right), \tag{3}$$

where w_i is the weight of the ith fuzzy judgment, $i = 1, 2, \ldots, p$. The weights usually satisfy the following normalization condition:

$$\sum_{i=1}^{p} w_i = 1. \tag{4}$$

Now our problem is how to determine the optimal weight w_i of the ith fuzzy judgment under the fuzzy GDM environment. Often, fuzzy judgments are largely dispersed and separated. In order to achieve the maximum similarity, fuzzy judgments should move towards one another. This is the principle on the basis of which an aggregated fuzzy judgment is generated. Based upon this principle, a least-square aggregation optimization approach is proposed to integrate fuzzy opinions produced by different DMs.

The generic idea of this proposed aggregation optimization approach is to minimize the sum of the squared distance from one fuzzy opinion to another and thus make them achieve maximum agreement. Specifically, the squared distance between $\tilde{Z}_i$ and $\tilde{Z}_j$ can be defined as

$$d_{ij}^2 = \left(\sqrt{(w_i \tilde{Z}_i - w_j \tilde{Z}_j)^2} \right)^2 = \sum_{l=1}^{3} \left(w_i z_{il} - w_j z_{jl} \right)^2. \tag{5}$$

Using this definition, we can construct the following optimization model, which minimizes the sum of the squared distances between all pairs of fuzzy judgments with weights:

$$\text{Minimize} \quad D = \sum_{i=1}^{p} \sum_{j=1, j\neq i}^{p} d_{ij}^2 = \sum_{i=1}^{p} \sum_{j=1, j\neq i}^{p} \left[\sum_{l=1}^{3} \left(w_i z_{il} - w_j z_{jl} \right)^2 \right] \tag{6}$$

$$\text{Subject to} \quad \sum_{i=1}^{p} w_i = 1 \tag{7}$$

$$w_i \geqslant 0, \quad i = 1, 2, \ldots, p. \tag{8}$$

In order to solve the above optimal weights, first, constraint (8) is not considered. If the solution turns out to be non-negative, then constraint (8) is satisfied automatically. Using the Lagrange multiplier method, Eqs. (6) and (7) in the above problem can construct the following Lagrangian function:

$$L(w, \lambda) = \sum_{i=1}^{p} \sum_{j=1, j\neq i}^{p} \left[\sum_{l=1}^{3} \left(w_i z_{il} - w_j z_{jl} \right)^2 \right] - 2\lambda \left(\sum_{i=1}^{p} w_i = 1 \right). \tag{9}$$

Differentiating (9) with w_i, we can obtain

$$\frac{\partial L}{\partial w_i} = 2 \sum_{j=1, j\neq i}^{p} \left[\sum_{l=1}^{3} \left(w_i z_{il} - w_j z_{jl} \right) z_{il} \right] - 2\lambda = 0 \quad \text{for each } i = 1, 2, \ldots, p. \tag{10}$$

Eq. (10) can be simplified as

$$(p-1) \left(\sum_{l=1}^{3} z_{il}^2 \right) w_i - \sum_{j=1, j\neq i}^{p} \left[\sum_{l=1}^{3} \left(z_{il} z_{jl} \right) \right] w_j - \lambda = 0 \quad \text{for each } i = 1, 2, \ldots, p. \tag{11}$$

Setting $W = (w_1, w_2, \ldots, w_p)^{\mathrm{T}}, I = (1, 1, \ldots, 1)^{\mathrm{T}}$ with the superscript T denoting the transpose, $b_{ij} = (p-1)\left(\sum_{l=1}^{3} z_{il}^2 \right)$, $i = j = 1, 2, \ldots, p, b_{ij} = -\sum_{l=1}^{3} \left(z_{il} z_{jl} \right), i, j = 1, 2, \ldots, p; j \neq i$ and

$$B = (b_{ij})_{p\times p} = \begin{bmatrix} (p-1)\left(\sum_{l=1}^{3} z_{1l}^2 \right) & -\sum_{l=1}^{3} \left(z_{1l} z_{2l} \right) & \cdots & -\sum_{l=1}^{3} \left(z_{1l} z_{pl} \right) \\ -\sum_{l=1}^{3} \left(z_{2l} z_{1l} \right) & (p-1)\left(\sum_{l=1}^{3} z_{2l}^2 \right) & \cdots & -\sum_{l=1}^{3} \left(z_{2l} z_{pl} \right) \\ \cdots & \cdots & \cdots & \cdots \\ -\sum_{l=1}^{3} \left(z_{pl} z_{1l} \right) & -\sum_{l=1}^{3} \left(z_{pl} z_{2l} \right) & \cdots & (p-1)\left(\sum_{l=1}^{3} z_{pl}^2 \right) \end{bmatrix}. \tag{12}$$

Using the matrix form and the above settings, Eqs. (11) and (7) can be rewritten as

$$BW - \lambda I = 0, \tag{13}$$

$$I^{\mathrm{T}} W = 1. \tag{14}$$

Similarly, Eq. (6) can be expressed in a matrix form as $D = W^{\mathrm{T}} B W$. Because D is a squared distance, which is usually larger than zero, B should be positive, definite and invertible. Using Eqs. (13) and (14) together, we can obtain

$$\lambda^* = 1/\left(I^{\mathrm{T}} B^{-1} I \right) \tag{15}$$

$$W^* = \left(B^{-1} I \right) / \left(I^{\mathrm{T}} B^{-1} I \right). \tag{16}$$

Since B is a positive definite matrix, all its principal minors will be strictly positive and thus B is a non-singular M-matrix (Berman and Plemmons, 1979). According to the properties of M-matrices, we know B^{-1} is non-negative. Therefore, $W^* \geqslant 0$, which implies that the constraint in Eq. (8) is satisfied.

After completing aggregation, a fuzzy group consensus can be obtained by Eq. (3). To obtain a crisp value of credit score, we use a defuzzification procedure to obtain the crisp value for decision-making purpose. According to Bortolan and Degani (1985), the defuzzified value of a triangular fuzzy number $\tilde{Z} = (z_1, z_2, z_3)$ can be determined by its centroid, which is computed by

$$z = \frac{\int_{z_1}^{z_3} x \mu_{\tilde{z}}(x) \mathrm{d}x}{\int_{z_1}^{z_3} \mu_{\tilde{z}}(x) \mathrm{d}x} = \frac{\int_{z_1}^{z_2} \left(x \cdot \frac{x - z_1}{z_2 - z_1} \right) \mathrm{d}x + \int_{z_2}^{z_3} \left(x \cdot \frac{z_3 - x}{z_3 - z_2} \right) \mathrm{d}x}{\int_{z_1}^{z_2} \left(\frac{x - z_1}{z_2 - z_1} \right) \mathrm{d}x + \int_{z_2}^{z_3} \left(\frac{z_3 - x}{z_3 - z_2} \right) \mathrm{d}x} = \frac{(z_1 + z_2 + z_3)}{3}. \tag{17}$$

So far, a final group consensus is computed with the above process. To summarize, the proposed intelligent-agent-based fuzzy GDM model is composed of five steps:

(1) To construct the GDM environment, some artificial intelligent techniques are first selected as intelligent agents.
(2) Based on the datasets, these selected intelligent agents, as group decision members, can produce different judgments by setting different parameters.
(3) For the different judgmental results, Eq. (2) is used to fuzzify the judgments of intelligent agents into fuzzy opinions.
(4) The fuzzy opinions are aggregated into a group consensus, using the above proposed optimization method, in terms of the maximum agreement principle.
(5) The aggregated fuzzy group consensus is defuzzified into a crisp value. This defuzzified value can be used as a final measurement for the final decision-making.

In order to illustrate and verify the proposed intelligent-agent-based fuzzy GDM model, the next section will present an illustrative numerical example and three real-world credit scoring experiments.

3. Experimental study

In this section, we first present an illustrative numerical example to explain the implementation process of the proposed fuzzy GDM model. Then three real-world credit scoring experiments are conducted; some interesting results are produced by comparison of these results with some existing methods.

3.1. An illustrative numerical example

To illustrate the proposed fuzzy GDM model, a simple numerical example is presented. Suppose the credit cutoff is 60 points; if the applicant's credit score is larger than this cutoff, then only his/her application will be accepted by the banks. According to the steps described in Section 2, we begin illustrating the implementation process of the proposed GDM model.

Suppose that there is a credit dataset, which is divided into two sets: training set and testing set. The training set is used to construct the intelligent agent models, while the testing set is used for verification purpose. In this example, three intelligent techniques, back-propagation neural network (BPNN) (Rumelhart et al., 1986), radial basis function network (RBFN) (Poggio and Girosi, 1990; Yu et al., 2006) and support vector machine regression (SVMR) (Vapnik, 1995; Xie et al., 2006), are employed as group members. The main reason for selecting these three intelligent techniques as agents is that they have good approximation capabilities. BPNN and RBFN are generally viewed as "universal approximators" (Hornik et al., 1989; White, 1990; Hartman et al., 1990; Park and Sandberg, 1991). In other words, these three models have the ability to provide flexible mapping between inputs and outputs and to give more accurate evaluation results than human experts because the intelligent agents can overcome the recognition bias and the subjectivity of human experts in GDM, as earlier noted in Section 1. Interested readers may please refer to Rumelhart et al. (1986), Poggio and Girosi (1990) and Vapnik (1995) for more details about the three intelligent techniques.

However, the performances of the intelligent agents are usually dependent on their architectures or some important parameters. As is known to all, neural networks are heavily dependent on the network topological structure and the support vector machines are heavily dependent on their selected kernel function and their parameters. For each model, we assume that ten different architectures or parameters are tried in the example. For this purpose, 30 different models are created. When the input information of a new applicant arrives, the 30 different models can provide 30 different credit scores for this new applicant. Assume that the 30 credit scores generated by BPNN, RBFN and SVMR agents are expressed as

$$f_{\mathrm{BPNN}} = (57.35, 54.76, 59.75, 60.13, 59.08, 61.24, 56.57, 58.42, 60.28, 55.85),$$
$$f_{\mathrm{RBFN}} = (58.86, 60.61, 59.81, 57.97, 61.31, 62.38, 60.79, 59.93, 61.12, 61.85),$$
$$f_{\mathrm{SVMR}} = (59.42, 60.33, 58.24, 61.36, 63.01, 60.85, 62.76, 61.79, 63.24, 62.66).$$

According to the previous setting, if the credit score is less than 60, the applicant will be rejected as a bad applicant. From the above credit scores of three DMs, the largest values of the scores from three agents are larger than 60 (i.e. the largest values of the BPNN, RBFN and SVMR agents are 61.24, 62.38 and 63.24, respectively). It seems that this new applicant will be accepted as a good applicant. Furthermore, according to the majority voting rule also, the application seems to be accepted because 17 out of 30 credit scores are larger than 60. However, the proposed fuzzy GDM model answers differently.

Using Eq. (2), evaluation results of the three intelligent agents (i.e. DMs) are fuzzified into three triangular fuzzy numbers, which are used as fuzzy opinions of the three DMs, i.e.

$$\tilde{Z}_{\mathrm{BPNN}} = (z_{\mathrm{BPNN1}}, z_{\mathrm{BPNN2}}, z_{\mathrm{BPNN3}}) = (54.76, 58.34, 61.24),$$
$$\tilde{Z}_{\mathrm{RBFN}} = (z_{\mathrm{RBFN1}}, z_{\mathrm{RBFN2}}, z_{\mathrm{RBFN3}}) = (57.97, 60.46, 62.38),$$
$$\tilde{Z}_{\mathrm{SVMR}} = (z_{\mathrm{SVMR1}}, z_{\mathrm{SVMR2}}, z_{\mathrm{SVMR3}}) = (58.24, 61.37, 63.24).$$

Then the subsequent work is to aggregate the three fuzzy opinions into a group consensus. Using the above optimization method, we can obtain the following results:

$$B = \begin{bmatrix} 20305 & -10522 & -10642 \\ -10522 & 21814 & -11032 \\ -10642 & -11032 & 22315 \end{bmatrix}, \quad B^{-1} = \begin{bmatrix} 0.2383 & 0.2299 & 0.2273 \\ 0.2299 & 0.2218 & 0.2193 \\ 0.2273 & 0.2193 & 0.2169 \end{bmatrix},$$
$$W^{*\mathrm{T}} = (0.3426, 0.3306; 0.3268), \quad \tilde{Z}^* = \sum_{i=1}^{3} w^* \tilde{Z}_i = (56.96, 60.03, 62.27)$$

The final step is to defuzzify the aggregated fuzzy opinion into a crisp value. Using Eq. (17), the defuzzified value of the final group consensus is calculated as follows:

$$z = (56.96 + 60.03 + 62.27)/3 = 59.75.$$

Because the credit score of the final group consensus is 59.75, the applicant should be rejected as a bad applicant. In order to verify the effectiveness of the proposed fuzzy GDM model, three real-world credit datasets are used.

3.2. Empirical comparisons with different credit datasets

In this subsection, three real-world credit datasets are used to test the effectiveness of the proposed intelligent-agent-based fuzzy GDM model. In the first dataset, we use different training sets to generate different evaluation results. In the second dataset, different evaluation results are produced by setting different model parameters. For the last dataset, the above two strategies are hybridized. For comparison purpose, we use two individual statistical models (linear regression – LinR and logistic regression – LogR) with three individual intelligent models (BPNN, RBFN and SVMR models with the best cross-validation performance); three intelligent ensemble models with majority voting rule (BPNN ensemble, RBFN ensemble and SVMR ensemble models) are also used to conduct the experiments. In addition, a majority-voting based GDM model integrating the three intelligent agents is also used for further comparison.

In addition, because the final goal of credit scoring is to support credit application decision, we classify applicants with credit scores higher than the cutoff as faithful customers and others as delinquent customers. To compare the performance of all the models considered in this study, we calculate the Type I accuracy, Type II accuracy and Total accuracy, which is expressed as

$$\text{Type I accuracy} = \frac{\text{number of classified and also observed as bad}}{\text{number of observed bad}}, \tag{18}$$
$$\text{Type II accuracy} = \frac{\text{number of classified and also observed as good}}{\text{number of observed good}}, \tag{19}$$
$$\text{Total accuracy} = \frac{\text{number of correct classifications}}{\text{number of total evaluations}}. \tag{20}$$

In order to rank all the models, we use the area under the receiver operating characteristic (ROC) graph (Fawcett, 2004) as another performance measurement. The ROC graph is a useful technique for ranking models and visualizing their performance. Usually, ROC is a two-dimensional graph in which *sensitivity* is plotted on the *Y*-axis and 1-*specificity* is plotted on the *X*-axis, as illustrated in Fig. 2. Actually, the *sensitivity* is equal to Type II accuracy and the *specificity* is equal to Type I accuracy.

Fig. 2 shows ROC curves of two different models, labeled *A* and *B*. To perform the model ranking task, a common method is to calculate the area under the ROC curve, abbreviated as AUC. Since the AUC is a portion of the area of the unit square, its value is always between 0 and 1. Fig. 2 shows the AUC of two different models with different fillings. Particularly, the AUC of Model *A* is the area of the skew line, while the AUC of model *B* is the area of the shaded part. Generally, a model with a large AUC will have a good average performance. For example, in this figure, the AUC of model *A* is larger than that of model *B*; thus the performance of model*A* is better than that of model*B*. However, it is possible for a large-AUC model to perform worse than a small-AUC model in a specific region of ROC space. Fig. 2 illustrates an example of this: model *A* is generally better than model*B*, except at (1-specificity) >0.7, where model *B* has a slight advantage. But AUC can well describe the general behavior of the classification model because it is independent of any cutoff or

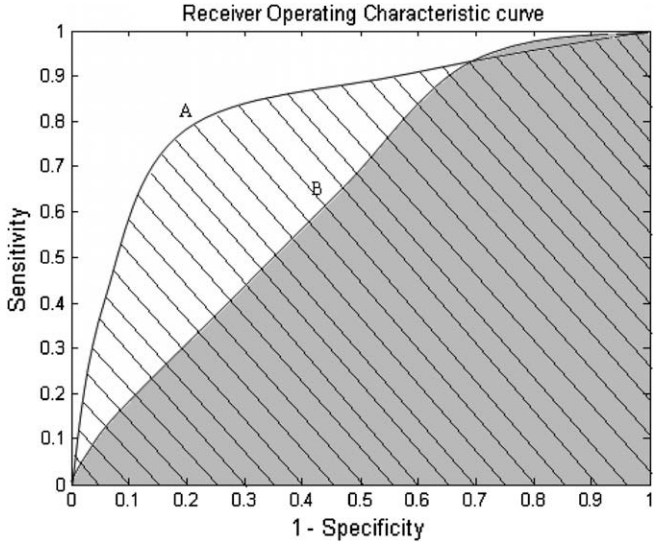

Fig. 2. ROC curve and AUC for two different models.

misclassification costs used for obtaining a class label. Due to this characteristic, it is widely used in practice. For AUC calculation, we use Algorithm 3 proposed by Fawcett (2004) in the following experiments.

3.2.1. Dataset I: England credit application example

The first credit dataset is from a financial service company of England, obtained from accessory CDROM of Thomas et al. (2002). The dataset includes detailed information of 1225 applicants, including 323 observed bad applicants. In the 1225 applicants, the number of good cases (902) is nearly three times that of bad cases (323). To make the numbers of the two classes near equal, we triple the number of bad cases, i.e. we add two copies of each bad case. Thus the total dataset grows to 1871 cases. The purpose of doing this is to avoid having too many good cases or too few bad cases in the training sample. Then we randomly draw 1000 cases comprising 500 good cases and 500 bad cases from the total of 1871 cases as the training samples and treat the rest as testing samples (i.e. 402 good applicants and 469 bad applicants).

To evaluate the applicant's credit score, 14 decision attributes are used as a set of decision criteria for credit scoring, which are described as follows:

(01) Year of birth.
(02) Number of children.
(03) Number of other dependents.
(04) Is there a home phone.
(05) Applicant's income.
(06) Applicant's employment status.
(07) Spouse's income.
(08) Residential status.
(09) Value of home.
(10) Mortgage balance outstanding.
(11) Outgoings on mortgage or rent.
(12) Outgoings on loans.
(13) Outgoings on hire purchase.
(14) Outgoings on credit cards.

Using this dataset and a set of evaluation criteria, we can construct an intelligent-agent-based fuzzy GDM model for multicriteria credit decision-making. The basic purpose of the GDM model is to make full use of group knowledge and intelligence. As mentioned earlier, group members in this study are some intelligent agents, rather than human experts. For simplicity, this study still uses three typical AI techniques, i.e. BPNN, RBFN and SVMR. That is, the three intelligent agents are seen as group members of GDM. From the above setting, a multicriteria GDM model for credit scoring can be shown in Fig. 3.

Applicants	DM_1 (BPNN)			DM_2 (RBFN)			DM_3 (SVMR)		
	x_1	…	x_{14}	x_1	…	x_{14}	x_1	…	x_{14}
1									
…									
N									

Fig. 3. A group decision table for credit scoring.

According to the previous setting at the beginning of Section 3.2, we use different training sets to generate different evaluation results, i.e. different credit scores. Here we use a typical data sampling algorithm – bagging algorithm (Breiman, 1996; Lai et al., 2006a) - to generate different training sets. Bagging is a widely used data sampling method in machine learning. Given that the size of the original data set DS is P, the size of the new training data is N, and the number of new training data items is m; the bagging sampling algorithm is shown in Fig. 4.

The bagging algorithm is very efficient in constructing a reasonable size of training set due to the feature of random sampling with replacement. Therefore, bagging is a useful data sampling method for machine learning (Breiman, 1996). In this study, we use the bagging algorithm to generate different training data subsets. Of course, besides the bagging algorithm, other data sampling approaches are also used. In this study, we use 20 different training sets (i.e. $P = 1871$, $N = 1000$, and $m = 20$) to create 20 different evaluation results for each intelligent agent. In the following, we describe some model settings of each intelligent agent.

In the BPNN model, a three-layer feed-forward BP network with seven TANSIG neurons in the hidden layer and one PURELIN neuron in the output layer is used. In this model, 14 decision attributes in the dataset are used as model inputs. The network training function is the TRAINLM (i.e. the core training algorithm is Levenberg–Marquardt algorithm, which is a fast learning algorithm for back-propagation network). Learning and momentum rates are set at 0.1 and 0.15 respectively. The accepted average squared error is 0.001 and the training epochs are 1000. The above parameters are obtained by the root mean squared error (RMSE) evaluation. To overcome the overfitting problem, the two-fold CV method is used. In the two-fold CV method, the first step is to divide the training dataset into two non-overlapping subsets. Then we train a BPNN, using the first subset of training data and validate the trained BPNN on the second subset. Subsequently, the second subset is used for training and the first subset is used for validation. Use of the two-fold CV is actually a reasonable compromise, considering the computational complexity of the systems. Furthermore, an estimate from the two-fold CV is likely to be more reliable than an estimate from a common practice using a single validation set.

In the RBFN model, we use the standard RBF neural network with seven hidden nodes and one output node. Gaussian radial basis function is used as the transfer function in hidden nodes. The cluster center and radius of Gaussian radial basis function is determined by average and standard deviations of the samples. In the SVMR model, the kernel function is Gaussian function with regularization parameters $C = 48$ and $\sigma^2 = 10$. Similarly, the above parameters are obtained by the grid search method. Because the three individual intelligent models are finally determined by the two-fold cross-validation technique, we use 500 samples as the first subset and the remaining 500 samples as the second subset, within the 1000 training samples. In addition, each of the three ensemble models utilize 20 different training sets generated by the bagging algorithm to create different ensemble members and then use the majority voting rule to aggregate the results of the ensemble members. For a majority of GDM models, sixty members produced by three intelligent agents are used to make final decisions via the majority voting principle. For the fuzzy GDM model, it is done by following the process described in Section 3.1.

```
Input: original data set DS
Output: The generated new training subsets {TR1, TR2, ..., TRm}
For t = 1 to m
  For i = 1 to N
    RandRow = P * rand ()
    If RandRow <= P
      Pt(i, AllColumns) = DS(RandRow, AllColumns)
    End If
  Next i
Next t
Output the final training subsets {TR1, TR2, ..., TRm}
```

Fig. 4. Bagging algorithm for data sampling.

According to the previous experiment design and model setting, the final computational results are shown in Table 1. As can be seen from Table 1, we can find the following conclusions.

(1) For the four evaluation criteria, the proposed intelligent-agent-based fuzzy GDM model performs the best, followed by the majority-based GDM and the three intelligent ensemble models; the individual BPNN model is the worst, indicating that the proposed fuzzy GDM model has a good generalization capability in credit risk evaluation. The reason that leads to this phenomenon comprises the following four aspects. First of all, aggregating multiple diverse decision-makers' (i.e. different intelligent agents in this study) knowledge into a group consensus can remedy the shortcomings of any individual decision-maker (i.e. individual AI agents here), thus increasing the decision reliability. Secondly, the proposed fuzzy GDM model utilizes approximations of both the inputs and the outputs within the GDM, as it fuzzified the inputs and defuzzified the output. Comparatively, intelligent ensemble models and individual methods do not use any such approximations. Third, the fuzzification processing of different prediction results can not only speed up the computational efficiency but also retain enough information for aggregation purpose. Fourth, the aggregation of different results can reduce the variance of generalization error and therefore produce a more robust result than the individual models. Finally, besides the internal diversity from every intelligent agent, the fuzzy GDM has an additional source of diversity not present in the ensemble model. The source of diversity is a mixture of decision makers, including BPNN, RBFN and SVMR agents. This extra diversity may help the proposed fuzzy GDM model to have a good generalization performance.
(2) In many empirical studies, like Wang et al. (2005) and Lai et al. (2006b), Type I accuracy should be worse than Type II accuracy because distinguishing a bad applicant is more difficult than classifying an applicant as good customer, to some extent. However, for the results of Type I and Type II accuracy reported in this study, we find that Type I accuracy is slightly higher than Type II accuracy, which is different from other prediction results. The main reason is that we create two copies of bad applicants in the sample and, therefore, some replications are labeled as bad applicants in testing, as previously mentioned.
(3) Of the five individual models, the SVMR model performs the best, followed by individual RBFN and logistic regression models. This shows that the SVMR model has good approximation capability for credit scoring. Surprisingly, the performance of the BPNN model is slightly worse than those of logistic regression and linear regression models. Because overfitting is avoided via cross-validation technique, the possible reason leading to that is that BPNN may encounter local minima problem.
(4) In the three intelligent ensemble models, the SVMR ensemble is the best. This conclusion is similar to the previous conclusion; it further confirms that the SVMR model is one of the best predictors. There are two main reasons. The first is that the SVMR adopts the structural risk minimization principle (Vapnik, 1995), which can overcome local minima problem. The second reason is that they can perform nonlinear mapping from an original input space into a high dimensional feature space. This helps it capture more non-linear information from original datasets and thus increases its classification capability.
(5) In all the intelligent models, an interesting finding is that the performance of the RBFN is consistently better than that of the BPNN. The main reasons are two-fold. On one hand, the RBFN model can overcome the local minima problem, which often occurs in the BPNN model. On the other hand, the parameters that need to be optimized lie only in the hidden layer of the RBFN model. Finding the parameters is only a solution of a linear problem and they are obtained through interpolation (Bishop, 1991). For this reason, the RBFN model can usually reach near perfect accuracy on the training data set without trapping into local minima (Chen et al., 1990; Wedding and Cios, 1996).

Table 1
Performance comparisons with different models for England dataset

Model	Type I (%) (Specificity)	Type II (%) (Sensitivity)	Total (%)	AUC
Individual LinR	65.25	61.19	63.38	0.6322
Individual LogR	65.46	61.69	63.72	0.6357
Individual BPNN	63.97	60.20	62.22	0.6208
Individual RBFN	70.79	66.67	68.89	0.6873
Individual SVMR	72.07	67.41	69.92	0.6974
BPNN ensemble	73.56	68.66	71.30	0.7111
RBFN ensemble	77.40	69.90	73.94	0.7365
SVMR ensemble	78.89	73.63	76.46	0.7626
Majority GDM	79.96	74.63	77.50	0.7729
Fuzzy GDM	***82.94***	***76.87***	***80.14***	***0.7990***

(6) It is worth noting that the majority-voting-based GDM model has also shown good prediction performance. Relative to the individual models and individual intelligent agent based ensemble models, the good performance of both the majority-based GDM model and the fuzzy GDM model mainly comes from aggregation of different information produced by different group members, rather than the group aggregation rule (i.e. majority voting rule and the fuzzy aggregation rule). Although the majority-based GDM model is slightly inferior to the fuzzy GDM model, the difference between the two is not significant when measured through the McNemar's test (see Section 3.2.4). One possible reason for this insignificant difference is that the fuzzy aggregation rule only provides a small portion of contribution to performance improvement of the proposed fuzzy GDM model; the main contribution to performance improvement of the proposed fuzzy GDM model comes from integration of diversity, as indicated in the first conclusion. But the real reasons leading to this slight difference are unknown, which is worth exploring further in the future.

3.2.2. Dataset II: Japanese credit card application example

The second dataset is about Japanese credit card application data obtained from UCI Machine Learning Repository (http://www.ics.uci.edu/~mlearn/databases/credit-screening/). For confidentiality, all attribute names and values have been changed to meaningless symbols. After deleting data with missing attribute values, we obtain 653 data, with 357 cases granted credit and 296 cases refused. To delete the burden of resolving multi-category cases, we use 13 attributes – A1–A5, A8–A15. Because we generally should substitute k-class attributes with k-1 binary attributes, which greatly increase the dimensions of the input space, we do not use two attributes: A6 and A7. In this empirical test we randomly draw 400 data from the 653 data as the training sample and the rest as the test sample. According to the previous setting at the beginning of Section 3.2, we use different parameters to generate different evaluation results, i.e. different credit scores, for each intelligent agent.

In the BPNN model, a three-layer feed-forward BP network with thirteen inputs and one output is used. To create different BPNN models, different numbers of hidden neurons are used. For consistency, we create 20 different BPNN models with different hidden neurons. That is, the number of hidden neurons varies from 6 to 25, with an increment of one. Similar to the first experiment, the network training algorithm is the Levenberg–Marquardt algorithm. Besides, the learning and momentum rates are set to 0.15 and 0.18 respectively. The accepted average squared error is 0.001 and the training epochs are 1200. The above parameters are obtained by the RMSE evaluation.

In RBFN, we use the same model as in the first experiment. That is, a standard RBF neural network with Gaussian radial basis function is used. Different from the first experiment, we vary the values of the cluster center and radius to create different RBFN models. For cluster center, ten different values (varying from 10 to 100 with an increment of ten) are used to construct 10 different RBFN models. Similarly, ten different radiuses (varying from 1 to 10 with an increment of one) are used to create 10 different RBFN models. Thus 20 different RBFN models are created and accordingly 20 different credit scores can be obtained from 20 different RBFN models.

In SVMR, the SVMR model with Gaussian kernel function is used. We use different regularization parameters C and σ^2 to create 20 different models. That is, we use ten different C and ten different σ^2 for different SVM models. Specifically, C varies from 10 to 100 with an increment of 10 and σ^2 is fixed to be 5; while σ^2 varies from 1 to 10 with an increment of one and C is fixed to be 50. In this way, 20 different models are generated and accordingly 20 different credit scores are obtained. Because the three individual intelligent models are finally determined by two-fold CV technique, we use 200 data as the first subset and the remaining 200 data as the second subset within the 400 training samples. In addition, each of the three ensemble models utilizes 20 different intelligent models with different parameters to create different ensemble members and then uses the majority voting rule to fuse the results of the ensemble members. For the fuzzy GDM model, it is done by following the process of Section 3.1. Table 2 summarizes the comparisons of the different models.

Table 2
Performance comparisons with different models for Japanese dataset

Model	Type I (%) (Specificity)	Type II (%) (Sensitivity)	Total (%)	AUC
Individual LinR	82.17	82.29	82.21	0.8222
Individual LogR	82.80	83.33	83.00	0.8307
Individual BPNN	80.89	81.25	81.03	0.8107
Individual RBFN	83.44	84.38	83.79	0.8391
Individual SVMR	78.98	82.29	80.24	0.8064
BPNN ensemble	81.25	83.44	82.21	0.8243
RBFN ensemble	83.44	85.42	84.18	0.8443
SVMR ensemble	80.25	82.29	81.02	0.8127
Majority GDM	84.71	85.42	84.98	0.8507
Fuzzy GDM	***85.99***	***86.46***	***86.17***	***0.8622***

Table 2 shows several interesting results, as illustrated below:

(1) It is not hard to find that the fuzzy GDM model achieves the best performance. The majority-vote-based GDM model and the RBFN ensemble model achieve the second and third best performances respectively.
(2) Of the five single models, the RBFN model performs the best, followed by single logistic regression and single linear regression. Surprisingly, the individual SVMR model performs the worst, which is distinctly different from the results of the first dataset. The reason for this is unknown and it is worth exploring further in future research. Although the performances of the single BPNN and the single SVMR model are worse than other three single models, the difference between them is insignificant according to the results of statistical test.
(3) In the three listed ensemble models, the SVMR ensemble performs worse than the other two ensemble models, i.e. BPNN ensemble and RBFN ensemble. The main reason is that single BPNN and RBFN are much better than the single SVMR model. Even individual logistic regression and the single RBFN model are also better than the SVMR ensemble model. This indicates that the ensemble model will perform poor if the performances of the single members constituting the ensemble are bad.
(4) Generally speaking, the proposed fuzzy GDM model performs the best in terms of Type I accuracy, Type II accuracy, Total accuracy and AUC, revealing that the proposed fuzzy GDM model is a feasible solution to improve the accuracy of credit risk evaluation.

3.2.3. Dataset III: German credit card application example

The German credit card dataset is provided by Professor Dr. Hans Hofmann of the University of Hamburg and is available at UCI Machine Learning Repository (http://www.ics.uci.edu/~mlearn/databases/statlog/german/). It contains 1000 data, with 700 cases granted credit card and 300 cases refused. In these instances, each case is characterized by 20 decision attributes, 7 numerical and 13 categorical, which are described as follows:

(01) status of existing checking account (categorical);
(02) duration in months (numerical);
(03) credit history (categorical);
(04) purpose (categorical);
(05) credit account (numerical);
(06) savings account/bonds (categorical);
(07) present employment since (categorical);
(08) installment rate in percentage of disposable income (numerical);
(09) Personal status and sex (categorical);
(10) other debtors/guarantors (categorical);
(11) present residence since (numerical);
(12) property (categorical);
(13) age in years (numerical);
(14) other installment plans (categorical);
(15) housing (categorical);
(16) number of existing credits at this bank (numerical);
(17) job (categorical);
(18) number of people being liable to provide maintenance for (numerical);
(19) have telephone or not (categorical); and
(20) foreign worker (categorical).

To make the numbers of the two classes near equal, we double bad cases, i.e. we add one copy of each bad case. Thus the total dataset now has 1300 cases. This processing is similar to the first dataset and the main reason of such a preprocessing step is to avoid drawing too many good cases or too few bad cases in the training sample. Then we randomly draw 800 data with 400 good cases and 400 bad cases from the 1300 data as the training sample and the remaining 500 cases are used as the testing sample (i.e. 300 good applicants and 200 bad applicants). According to the previous setting at the beginning of Section 3.2, we use bagging sampling algorithms to create 10 different training sets. At the same time, 10 different parameters for each intelligent agent are used to create different models for the third dataset. In this way, 20 different models for each intelligent agent are produced. Accordingly, different evaluation results, i.e. different credit scores, are generated from each intelligent agent. Because the 20 different models are created by using such a hybrid strategy, the basic settings of each intelligent agent model are similar to the previous two datasets and are omitted here because of space consideration. Similar to the second dataset, the two-fold cross-validation technique uses 400 data as

the first subset and the remaining 400 data as the second subset, within the 800 data in the training sample. In addition, each of the three ensemble models utilizes 20 different intelligent models with different training sets and different parameters to create different ensemble members and then uses the majority voting rule to integrate the results of ensemble members. For the fuzzy GDM model, it is done by following the process described in Section 3.1. Similar to the above two datasets, the final computational results are shown in Table 3.

Comparing Tables 1 and 3, some similar conclusions are obtained. Particularly, this dataset again confirms that the proposed fuzzy GDM model is suitable for credit risk evaluation task, implying that it is a very promising solution to financial multicriteria decision-making problem. A visualized explanation for performance comparison with different models is illustrated with ROC curve in Fig. 5.

3.2.4. Further discussions

The above illustrative example, provided in Section 3.1, explains the implementation process of the proposed fuzzy GDM methodology and the subsequent three practical datasets verify the effectiveness of the proposed method. Through accuracy and AUC measurements, we can judge which model is the best and which model is the worst. However, it is unclear what the differences between good models and bad ones are. For this, we conducted McNemar's test (McNemar, 1947) to examine whether the proposed fuzzy GDM model significantly outperforms the other nine models listed in this study. As a non-parametric test for two related samples, it is particularly useful for before–after measurement of the same subjects (Cooper and Emory, 1995). Taking the first dataset as an example, Table 4 shows the results of the McNemar's test for England credit dataset to statistically compare the performance in respect of testing data among the ten models. For space consideration, the results on McNemar's test for other two practical datasets are omitted here. Actually, we can obtain some similar conclusions from the second and third datasets via McNemar's test. Note that the results listed in Table 4 are the Chi squared values and p values are in brackets.

Table 3
Comparison of performances of different models for the German dataset

Model	Type I (%) (Specificity)	Type II (%) (Sensitivity)	Total (%)	AUC
Individual LinR	71.50	62.33	66.00	0.6692
Individual LogR	77.50	69.00	72.40	0.7325
Individual BPNN	75.00	67.33	70.40	0.7117
Individual RBFN	78.50	71.00	74.00	0.7475
Individual SVMR	80.50	74.67	77.00	0.7758
BPNN ensemble	81.00	73.67	76.60	0.7733
RBFN ensemble	82.00	75.33	78.00	0.7867
SVMR ensemble	82.50	77.33	79.40	0.7992
Majority GDM	83.00	79.00	80.60	0.8100
Fuzzy GDM	***84.50***	***80.33***	***82.00***	***0.8242***

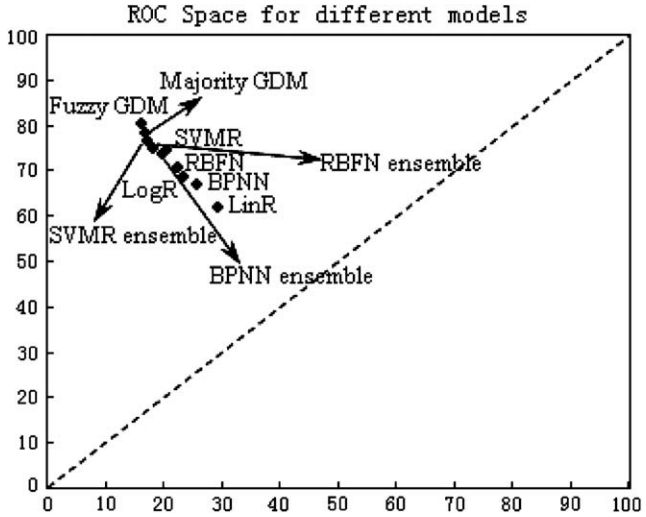

Fig. 5. A graphic performance comparison for different models in German dataset.

Table 4
McNemar's test for pairwise comparison of performance

Model	Majority GDM	SVMR ensemble	RBFN ensemble	BPNN ensemble	Individual SVMR	Individual RBFN	Individual LogR	Individual LinR	Individual BPNN
Fuzzy GDM	1.3120 (0.2521)	2.5420 (0.1108)	7.0230 (0.0080)	13.655 (0.0002)	17.802 (0.0000)	21.191 (0.0000)	41.235 (0.0000)	42.734 (0.0000)	47.859 (0.0000)
Majority GDM		0.1600 (0.6895)	2.1280 (0.1447)	6.2980 (0.0121)	9.2250 (0.0024)	11.726 (0.0006)	27.658 (0.0000)	28.901 (0.0000)	33.189 (0.0000)
SVMR ensemble			1.0210 (0.3123)	4.2550 (0.0391)	6.7150 (0.0096)	8.8760 (0.0029)	23.225 (0.0000)	24.368 (0.0000)	28.331 (0.0000)
RBFN ensemble				1.0150 (0.3138)	2.3640 (0.1242)	3.7130 (0.0540)	14.262 (0.0002)	15.167 (0.0001)	18.347 (0.0000)
BPNN ensemble					0.2360 (0.6269)	0.7680 (0.3809)	7.4650 (0.0063)	8.1270 (0.0044)	10.508 (0.0012)
Individual SVMR						0.1200 (0.7290)	4.8600 (0.0275)	5.3980 (0.0202)	7.3710 (0.0066)
Individual RBFN							3.2980 (0.0694)	3.7440 (0.0530)	5.4150 (0.0200)
Individual LogR								0.0060 (0.9367)	0.2230 (0.6366)
Individual LinR									0.1250 (0.7237)

As shown in Table 4, we can draw the following conclusions:

(1) The proposed fuzzy GDM model outperforms the RBFN ensemble, BPNN ensemble, individual SVMR, RBFN, BPNN, LogR and LinR models at 1% statistical significance level. However, the proposed GDM model does not significantly outperform the majority-vote-based GDM model and the SVMR ensemble model.
(2) For the majority-vote-based GDM model, we can find that the majority-vote-based GDM model outperforms all the five individual models (i.e., individual SVMR, RBFN, BPNN, LogR, and LinR models) at 1% significance level. Similarly, it is better than the BPNN ensemble model at 5% significance level, but the McNemar's test does not conclude that it performs better than the SVMR ensemble and RBFN ensemble models.
(3) Similar to the majority-vote-based GDM model, the SVMR ensemble model can also outperform all the five individual models at 1% significance level and it can also perform better than the BPNN ensemble model at 5% significance level. Interestedly, it does not outperform the RBFN ensemble model at 10% significance level.
(4) For the RBFN ensemble model, it can outperform the individual BPNN, LogR and LinR models at 1% significance level and it performs better than the RBFN model at 10% significance level. However, the RBFN ensemble model does not outperform the BPNN ensemble model and the individual SVMR model at 10% significance level. Similarly, the BPNN ensemble model leads to a similar finding.
(5) For the individual SVMR model, it cannot outperform the RBFN model from Table 4, but it is easy to find that it performs better than the individual BPNN, LogR and LinR models at 5% significance level. For the individual RBFN model, it can outperform the remaining three individual models at 10% significance level. In addition, Table 4 also shows that the performances of the individual BPNN, LogR and the LinR models do not differ significantly from each other. All findings are consistent with results reported in Table 1. For the second and third datasets, we can draw some similar conclusions, as previously mentioned.

Besides the differences among the different models, there is a conflicting viewpoint about the model performance improvement. It is the famous "no free lunch" theorem of the machine learning theory (Schaffer, 1994; Wolpert and Macready, 1997). Roughly speaking, these theorems say that no model (i.e. predictor or classifier) can outperform another on average, over all possible classification problems, and implicitly question the utility of learning research. However, as Rao et al. (1995) have shown, this theorem does not necessarily apply to every case because not all classification problems are equal. Recently, Domingos (1998) proposed a simple cost model to prove the possibility of getting a free lunch in machine learning applications. Suppose there are two classifiers C_1 and C_2; classifier C_2 will have a globally better performance than classifier C_1 if generalization accuracy of C_2 is better than that of C_1 in the problem domains, as illustrated in Fig. 6. Note that the shaded area represents the accuracy gained at no cost.

In Fig. 6, C_1 and C_2 follow the "no free lunch" theorem because they both have an average accuracy of 50% over all domains. However, C_2 has a higher average effective performance than C_1, since only the area above A_0 counts for purposes of computing effective performance. In short, a good strategy for research is to keep improving the current classifiers in the domains where they do well, regardless of the fact that this makes them worse where they perform poorly. Not surprisingly, this is largely what is done in practice (Domingos, 1998). Due to such a fact, we have enough reasons to believe that our proposed fuzzy GDM model can outperform other models listed in this study. That is, in this study, the "no free lunch" theorem does not apply because not all credit classification problems are handled equally. Meanwhile, three real-world experiments also confirm that the proposed fuzzy GDM model can effectively improve credit classification

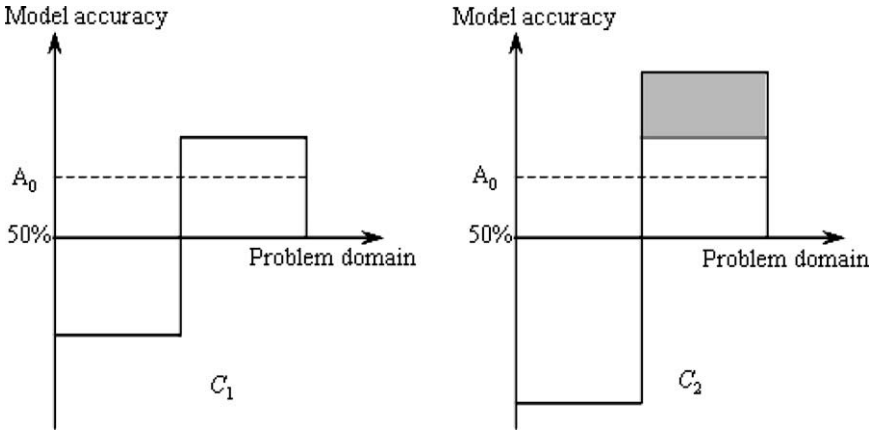

Fig. 6. Improving the global performance of a classifier.

performance relative to other classification models listed in this study. This also implies that our proposed model can be used as an alternative solution to credit risk evaluation problems. For further information about getting a free lunch for machine learning applications, interested readers can refer to Rao et al. (1995) and Domingos (1998) for more details.

4. Conclusions

In this study, a novel intelligent-agent-based fuzzy GDM model is proposed as a financial multicriteria decision-making (MCDM) tool to support credit scoring problems. Different from commonly used "one-member-one-vote" or "majority-voting-rule" ensemble models, the novel fuzzy GDM model first uses several intelligent agents to evaluate the customer over a number of criteria, then the evaluation results are fuzzified into some fuzzy judgments, and finally these fuzzy judgments are aggregated and defuzzified into a group consensus as a final group decision measurement.

For illustration and verification purposes, an illustrative example is used to show the implementation process of the proposed fuzzy GDM model; three publicly available credit datasets have been used to test the effectiveness and decision power of the proposed fuzzy GDM approach. All results reported in the three experiments clearly show that the proposed fuzzy GDM model can outperform other comparable models, including five single models and three majority-voting-based intelligent ensemble models, as well as the majority-based GDM model. These results reveal that the proposed fuzzy GDM model can provide a promising solution to credit scoring tasks, implying that the proposed fuzzy GDM technique has a great potential for being applied to other financial MCDM problems.

However, it is worth noting that the classification accuracy used in the proposed fuzzy GDM model is also influenced by the overlap in the way the range of some evaluation results is split into various categories (e.g., range of values for small, medium and large). Again, these are the pitfalls associated with mechanisms used for both fuzzification and defuzzification of input and output data (Piramuthu, 1999). Furthermore, this work can be extended easily into the case of trapezoidal fuzzy numbers and thus it can be applied to more financial MCDM problems. In addition, in credit scoring system, the credit-granting institutions often need to be able to provide some specific information of why credit was refused to an applicant. But our proposed approach does not give any insight into the logics of the decision model. Therefore, using these intelligent agents to extract the decision rules (Craven and Shavlik, 1994; Andrews et al., 1995; Martens et al., 2007) or determine some key decision attributes might also be an interesting topic for future credit scoring research. We will look into these issues in the future.

Acknowledgements

The authors would like to thank the guest editor and five anonymous referees for their valuable comments and suggestions. Their comments helped improve the quality of the paper immensely. This work is supported by Grants from the National Natural Science Foundation of China (NSFC Nos. 70221001, 70601029), the Knowledge Innovation Program of the Chinese Academy of Sciences (CAS No. 3547600, 3046540, 3047540), the Academy of Mathematics and Systems Science (AMSS No. 3543500) of CAS and Strategic Research Grant of City University of Hong Kong (SRG Nos. 7001677, 7001806).

References

Andrews, R., Diederich, J., Tickle, A.B., 1995. Survey and critique of techniques for extracting rules from trained artificial neural networks. Knowledge-Based Systems 8 (6), 373–389.

Berman, A., Plemmons, R.J., 1979. Nonnegative Matrices in the Mathematical Sciences. Academic, New York.

Beynon, M.J., 2005. A method of aggregation in DS/AHP for group decision-making with the non-equivalent importance of individuals in the group. Computers & Operations Research 32, 1881–1896.

Bishop, C.M., 1991. Improving the generalization properties of radial basis function neural networks. Neural Computation 3, 579–588.

Bortolan, G., Degani, R., 1985. A review of some methods for ranking fuzzy subsets. Fuzzy Sets and Systems 15, 1–19.

Breiman, L., 1996. Bagging predictors. Machine Learning 26, 123–140.

Chen, M.C., Huang, S.H., 2003. Credit scoring and rejected instances reassigning through evolutionary computation techniques. Expert Systems with Applications 24, 433–441.

Chen, S., Billings, S.A., Cowan, C.F.N., Grant, P.M., 1990. Nonlinear systems identification using radial basis functions. International Journal of Systems Science 21, 2513–2539.

Cholewa, W., 1985. Aggregation of fuzzy opinions—an axiomatic approach. Fuzzy Sets and Systems 17, 249–258.

Cooper, D.R., Emory, C.W., 1995. Business Research Methods. Irwin, Chicago.

Craven, M.W., Shavlik, J.W., 1994. Using sampling and queries to extract rules from trained neural networks. In: Proceedings of the 11th International Conference on Machine Learning, New Brunswick, NJ, pp. 37–45.

Delgado, M., Herrera, F., Herrera-Viedma, E., Martinez, L., 1998. Combining numerical and linguistic information in group decision making. Information Sciences 107, 177–194.

DeSanctis, G., Gallupe, R.B., 1987. A foundation for the study of group decision support systems. Management Sciences 33 (5), 589–609.

Domingos, P., 1998. How to get a free lunch: A simple cost model for machine learning applications. In: Proceedings of AAAI-98/ICML-98 Workshop on the Methodology of Applying Machine Learning, 1-7. Madison, WI.
Fawcett, T., 2004. ROC graphs: Notes and practical considerations for researchers. Intelligent Enterprise Technologies Laboratory, HP Laboratories Palo Alto, HPL-2004-03.
Fisher, R.A., 1936. The use of multiple measurements in taxonomic problems. Annals of Eugenics 7, 179–188.
Glover, F., 1990. Improved linear programming models for discriminant analysis. Decision Science 21, 771–785.
Grablowsky, B.J., Talley, W.K., 1981. Probit and discriminant functions for classifying credit applicants: A comparison. Journal of Economic Business 33, 254–261.
Hartman, E.J., Keeler, J.D., Kowalski, J.M., 1990. Layer neural networks with Gaussian hidden units as universal approximations. Neural Computation 2 (2), 210–215.
Henley, W.E., Hand, D.J., 1996. A *k*-NN classifier for assessing consumer credit risk. Statistician 45, 77–95.
Hornik, K., Stinchocombe, M., White, H., 1989. Multilayer feedforward networks are universal approximators. Neural Networks 2, 359–366.
Huang, Z., Chen, H.C., Hsu, C.J., Chen, W.H., Wu, S.S., 2004. Credit rating analysis with support vector machines and neural networks: a market comparative study. Decision Support Systems 37, 543–558.
Irion, A., 1998. Fuzzy rules and fuzzy functions: a combination of logic and arithmetic operations for fuzzy numbers. Fuzzy Sets and Systems 99, 49–56.
Lai, K.K., Yu, L., Huang, W., Wang, S.Y., 2006a. A novel support vector machine metamodel for business risk identification. Lecture Notes in Artificial Intelligence 4099, 980–984.
Lai, K.K., Yu, L., Wang, S.Y., Zhou, L.G., 2006b. Credit risk analysis using a reliability-based neural network ensemble model. Lecture Notes in Computer Science 4132, 682–690.
Lai, K.K., Yu, L., Zhou, L.G., Wang, S.Y., 2006c. Credit risk evaluation with least square support vector machine. Lecture Notes in Artificial Intelligence 4062, 490–495.
Lai, K.K., Yu, L., Zhou, L.G., Wang, S.Y., 2006d. Neural network metalearning for credit scoring. Lecture Notes in Computer Science 4113, 403–408.
Lee, H.S., 2002. Optimal consensus of fuzzy opinions under group decision making environment. Fuzzy Sets and Systems 132, 303–315.
Lee, T.S., Chiu, C.C., Lu, C.J., Chen, I.F., 2002. Credit scoring using the hybrid neural discriminant technique. Expert Systems with Application 23 (3), 245–254.
Makowski, P., 1985. Credit scoring branches out. Credit World 75, 30–37.
Malhotra, R., Malhotra, D.K., 2002. Differentiating between good credits and bad credits using neuro-fuzzy systems. European Journal of Operational Research 136, 190–211.
Malhotra, R., Malhotra, D.K., 2003. Evaluating consumer loans using neural networks. Omega 31, 83–96.
Mangasarian, O.L., 1965. Linear and nonlinear separation of patterns by linear programming. Operations Research 13, 444–452.
Martens, D., Baesens, B., Van Gestel, T., Vanthienen, J., 2007. Comprehensible credit scoring models using rule extraction from support vector machines. European Journal of Operational Research 183, 1466–1476.
McNemar, Q., 1947. Note on the sampling error of differences between correlated proportions and percentages. Psychometrica 12, 153–157.
Park, J., Sandberg, I.W., 1991. Universal approximation using radial basis function networks. Neural Computation 3 (2), 246–257.
Park, K.S., Kim, S.H., 1996. A note on the fuzzy weighted additive rule. Fuzzy Sets and Systems 77, 315–320.
Piramuthu, S., 1999. Financial credit-risk evaluation with neural and neurofuzzy systems. European Journal of Operational Research 112, 310–321.
Poggio, T., Girosi, F., 1990. Network for approximation and learning. Proceedings of the IEEE 78, 1481–1497.
Ramakrishnan, R., Rao, C.J.M., 1992. The fuzzy weighted additive rule. Fuzzy Sets and Systems 46, 177–187.
Rao, R.B., Gordon, D., Spears, W., 1995. For every action, is there really an equal and opposite reaction? Analysis of the conservation law for generalization performance. In: Proceeding of the Twelfth International Conference on Machine Learning, 471–479. Morgan Kaufmann: Tahoe City, CA.
Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning representations by back-propagating errors. Nature 323, 533–536.
Schaffer, C., 1994. A conservation law for generalization performance. Proceedings of the Eleventh International Conference on Machine Learning. Morgan Kaufmann, New Brunswick, New York, pp. 259–265.
Smalz, R., Conrad, M., 1994. Combining evolution with credit apportionment: a new learning algorithm for neural nets. Neural Networks 7, 341–351.
Thomas, L.C., 2002. A survey of credit and behavioral scoring: Forecasting financial risk of lending to consumers. International Journal of Forecasting 16, 149–172.
Thomas, L.C., Edelman, D.B., Crook, J.N., 2002. Credit Scoring and its Applications. Society of Industrial and Applied Mathematics, Philadelphia.
Thomas, L.C., Oliver, R.W., Hand, D.J., 2005. A survey of the issues in consumer credit modelling research. Journal of the Operational Research Society 56, 1006–1015.
Van Gestel, T., Baesens, B., Garcia, J., Van Dijcke, P., 2003. A support vector machine approach to credit scoring. Bank en Financiewezen 2, 73–82.
Vapnik, V.N., 1995. The Nature of Statistical Learning Theory. Springer, New York.
Varetto, F., 1998. Genetic algorithms applications in the analysis of insolvency risk. Journal of Banking and Finance 22, 1421–1439.
Wang, Y.Q., Wang, S.Y., Lai, K.K., 2005. A new fuzzy support vector machine to evaluate credit risk. IEEE Transactions on Fuzzy Systems 13, 820–831.
Wedding II, D.K., Cios, K.J., 1996. Time series forecasting by combining RBF networks, certainty factors and the Box-Jenkins model. Neurocomputing 10, 149–168.
White, H., 1990. Connectionist nonparametric regression: multilayer feedforward networks can learn arbitrary mappings. Neural Networks 3, 535–549.
Wiginton, J.C., 1980. A note on the comparison of logit and discriminant models of consumer credit behaviour. Journal of Financial Quantitative Analysis 15, 757–770.
Wolpert, D.H., Macready, W.G., 1997. No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation 1, 67–82.
Xie, W., Yu, L., Xu, S.Y., Wang, S.Y., 2006. A new method for crude oil price forecasting based on support vector machines. Lecture Notes in Computer Science 3994, 444–451.
Xu, Z.S., 2004. A method based on linguistic aggregation operators for group decision making with linguistic preference relations. Information Sciences 166, 19–30.
Xu, Z.S., 2005. Uncertain linguistic aggregation operators based approach to multiple attribute group decision making under uncertain linguistic environment. Information Sciences 169, 171–184.
Yager, R.R., 1993. A general approach to criteria aggregation using fuzzy measures. International Journal of Man–Machine Studies 39, 187–213.
Yager, R.R., 1994. Aggregation operators and fuzzy systems modeling. Fuzzy Sets and System 67, 129–145.

Yu, L., Huang, W., Lai, K.K., Wang, S.Y., 2006. A reliability-based RBF network ensemble model for foreign exchange rates predication. Lecture Notes in Computer Science 4234, 380–389.

Zhang, G., Lu, J., 2003. An integrated group decision-making method dealing with fuzzy preferences for alternatives and individual judgments for selection criteria. Group Decision and Negotiation 12, 501–515.