Sigma J Eng & Nat Sci 37 (3), 2019, 855-870

Sigma Journal of Engineering and Natural Sciences
Sigma Mühendislik ve Fen Bilimleri Dergisi

Research Article

CREDIT SCORING BY ARTIFICIAL NEURAL NETWORKS BASED CROSS-ENTROPY AND FUZZY RELATIONS

Damla ILTER[1], Ozan KOCADAGLI*[2]

[1]*Department of Statistics, Mimar Sinan Fine Arts University, ISTANBUL;* ORCID: 0000-0002-9844-4616
[2]*Department of Statistics, Mimar Sinan Fine Arts University, ISTANBUL;* ORCID: 0000-0003-4354-7583

Received: 22.11.2018 Revised: 31.01.2019 Accepted: 28.03.2019

ABSTRACT

The credit scoring is one of the major activities in the banking sector. Because of growing market and increasing the loan applications, this field still continues its concern in terms of rating the applicants and assessing the credit amounts. To reduce the number of wrong decisions in the credit evaluation process, the decision makers focus on estimating more robust models. However, the traditional methods are criticized due to various pre-requisites and linear approximations in the high dimensional and excessive nonlinear cases. For this reason, artificial intelligence techniques are mostly preferred to handle the credit scoring problems accurately. This study presents an efficient procedure that is based on ANNs with cross-entropy and fuzzy relations in the context of the credit scoring. In the implementations, the proposed procedure is applied to a couple of benchmark credit scoring data sets and its performance is compared with traditional approaches.

Keywords: Credit scoring, artificial neural networks, cross-entropy, fuzzy relations, gradient based algorithms.

1. INTRODUCTION

The value of revolving credit outstanding in the whole world economy has grown rapidly depending on many economic factors. Today, the financial institutions still continue to diversify the credit products to reach more customers in the context of the economic perspective. Apparently, this intention brings about various challenges such as evaluating the risk levels and credit scores of customers, assessing the credit amounts, the follow-up on payments, the legal processes, etc. Apart from the credit scoring and risk evaluation, most of workloads are directly interested in the management issues that can be accomplished via the corporate governance structure thoroughly. However, the financial institutions make a special effort for the systematic risk related to the credit scoring. Essentially, the credit scoring is an efficient method to measure the systematic risk when financing the individual customers as well as the small and medium-sized enterprises (SMEs). For this reason, the researchers and experts focus on improving more efficient procedures for the credit scoring process in terms of estimating more robust models using the statistical, operations research, artificial intelligence (AI), machine learning and hybrid approaches in the automated and consistent manner. Generally, the credit scoring problem deals

[*] Corresponding Author: e-mail: ozan.kocadagli@msgsu.edu.tr, tel: (212) 246 00 11 /5511

with two categories: “application scoring” and “behavioral scoring”. The former is interested in classifying the credit applicants into various risk groups and making a decision about whether any credit application is worthy to approve or not, but the latter deals with making an inference using the payment histories of customers [1]. This inference includes the critical strategies about credit limits of costume rs, payment difficulties, bankruptcy, etc. Besides, these inferential models also predict future purchasing behavior or credit status of customers [2].

Generally, it´s not easy to determine which variables should be handled to evaluate the credit risks since this kind of information is rarely shared by financial institutes. However, the modelling structure can be learned from our experience of financial markets, and more implicitly from the Fair, Isaac and Company (FICO) method, VantageScore, Ficth, Moody’s and Standart & Poor’s Ratings. Today, the FICO model or its different versions are still used by the majority of financial institutions. Even so, the financial institutes still continue to develop their credit scoring procedures depending on the number of customers, economic factors, industrial field being used the credit, etc., because the developed models should meet all the demands in this dynamical and stochastic environment.

Today’s technology allows the supercomputers to make more data processing due to more powerful processor and RAM capacities. Hence, this development prompts the researchers to use more efficient decision support systems those will be able to cope with big data sets as well as model complexity. In the literature, the researchers mostly have focused on hybridizing artificial neural networks (ANNs), support vector machines (SVM), regression trees, expert systems with meta-heuristic algorithms, Bayesian and fuzzy theories. Specifically, the deep learning which can be considered as an extension of ANNs where there are much larger layer sets and various combination activation functions is very popular research field nowadays. Even so, in the context of credit scoring literature, there are different approaches such as Discriminant Analysis [3,4], logistic regression analysis [5], Decision Tree [6,7,8,9], linear and nonlinear programming, decision support systems, ANNs, SVMs [5,10,11,12, 13, 14,15,16,17], Bayesian Networks [17,18, 19], Expert Systems and hybrid approaches [3,5,20,21,22,23,24,25], etc.

Recently the ensemble methods are also applied to credit scoring procedure [5], [7], [15], [16], [21], [26], [27], [28], [29], [30], [31], [32], [33]. Generally, these procedures present different ensemble methods such as AdaBoost, Bagging, Random Subspace, Stacking, Decorate and Rotation Forest with the following base classifiers: 1-nearest neighbor, LogR, multilayer perceptron, radial basis function, gamma, clustering-launched classification, SVM and C4.5 decision tree.

This study introduces to a novel procedure that allows the experts to estimate the efficient credit scoring models. Specifically, this procedure hybridizes the fuzzy relations with ANNs based cross-entropy and information criteria. While the fuzzy relation is used for reducing the dimension of feature matrix, ANNs present a natural and flexible way to model the high-dimensional and excessive non-linear systems without any restrictive assumption [34,35,36,37]. Thus, they are capable of give superior performance to the classical approaches. Also, they can be hybridized with other artificial intelligent techniques practically [36]. Therefore, ANNs are widely used in credit scoring problems because of their flexible structures [1,7,22,29,39,40,41,42,43,44,45,46,47].

2. MOTIVATION AND OVERVIEW

According to the literature, the hybrid approaches provide superior performance against the classical techniques. However, they bring out many shortcomings such as the control of tuning parameters, model complexity, time-consuming, processor and memory allocation, etc. In the context of ANNs, the researches suffer from the selection of risk function, model complexity, early-stopping, cross-validation, time-consuming [34,35]. Generally, they intend to minimize Mean Squared Error (MSE) at the classification problems such as the credit scoring. However, this approach is not inherently convenient for classification problem, because the output vector of ANNs consists of categorical variables. Specifically, the case of binary number {0, 1} denotes whether any credit application is worthy to approve or not [1,48]. To get rid of this inconvenience, various classifiers can be used such as cross-entropy, Kullback Leibler divergence, Bayesian classifiers and information criteria instead of MSE [29,38,40,43,49,50]. In this study, as a classifier in ANNs, the cross-entropy measure is preferred, because it is more convenient to represent a binary structure such as whether any credit application is worthy to approve or not.

To estimate a robust model by ANNs, another issue is complexity which is closely related to the over/lower fitting to training data [51,52]. For this reason, reducing the dimension of feature matrix plays important role to control the model complexity as well as determining the number of neurons in the hidden layer(s). However, the model complexity is mostly overlooked by researchers due to time consumption. In this study, to reduce the dimension of feature matrix, the fuzzy relations are proposed against traditional feature selection methodologies. This approach allows using various metrics to create a relation matrix among features. Thus, the equivalence classes over the fuzzy relations can be defined by means of various fuzzy composition operators where these classes consist of certain feature(s) with same characteristics. In this perspective, each class can be considered as a component or main factor as similar to the principal component analysis and factor analysis [38].

To determine the efficient number of neurons in the hidden layer, information criteria is a practical approach; otherwise the trial and error apparently causes waste of time [40,53,54,55,56]. For this reason, in the proposed procedure, the number of neuron is determined by Akaike Information Criterion (AIC), Corrected AIC (AICc) and Bayesian Information Criterion (BIC). In addition, to control the model complexity, early stopping approach based on the cross-validation is used.

In the context of training ANNs, the best practical way is to use gradient based algorithms such as Quasi-Newton known as Broyden, Fletcher, Goldfarb, and Shanno (BFGS), Levenberg-Marquardt (L-M), Scaled Conjugate Gradient (SCG), Gradient Descent (GD) and GD with momentum (GDwM). Particularly, BFGS utilizes the approximated Hessian matrix and update it each iteration instead its original one. This configuration allows solving the sophisticated unconstrained problems. L-M algorithm runs a non-negative damping parameter where its larger values make the algorithm closer to Gauss-Newton Method whereas its smaller values make it closer the GD. By means of efficient damping parameters, L-M is capable of very fast searching in the high dimensional parameter cases and non-linear least square problems. By using a step size scaling mechanism SCG avoids time consuming for line-search per learning iteration, thus it makes the algorithm much faster. GD minimizes the error function by performing a search using the learning rate parameter in the direction of the gradient vector through the solution space. However, updating the learning rate at each iteration is not practical and efficient approach for the high dimensional systems like ANNs. In addition, this algorithm mostly stuck in the local optima due to the use of gradient vector. To overcome this situation, GDwM can be preferred. Unlike GD, this algorithm utilizes a momentum constant together with the learning rate at each iteration; thus it gains a feature that is not trapped to the local optima [35,58,60]. In the light of the above, examining the pros and cons of gradient based algorithms plays important role to estimate the models from ANNs in the most accurate and best manner. For this reason, training ANNs via various gradient based algorithms provides an efficiency for the estimation procedure.

To summarize, the main purpose of this paper is to introduce an efficient approach that allows the decision makers to estimate the robust classification models for the credit scoring using ANNs based cross entropy and information criteria as well as fuzzy relations. Thus, the decision makers will be able to make more accurate decision about the credit approvals of the individual customers. Also, this procedure can be easily applied to another credit data set as well. To present the proposed approach, this paper is constructed as following. Section 3 introduces the fuzzy relation methodology in the context of reducing the dimension of feature matrix. Section 4 deals with the architecture of ANNs based cross-entropy and information criteria. Section 5 includes the credit scoring implementations where two benchmark data sets are handled. Section 6 is allocated for the results of analysis. Finally, the analysis results and outputs are discussed and interpreted in detail in the Section 7.

3. REDUCTION THE DIMENSIONALITY OF FEATURES USING FUZZY RELATIONS

3.1.1. Fuzzy Cartesian Products and Relations

Let $\tilde{A}_1, \tilde{A}_2, \dots, \tilde{A}_p$ be fuzzy sets in universes, $X_1, X_2, \dots, X_p$, respectively. Thus, Cartesian product $\tilde{A}_1 \times \tilde{A}_2 \times \dots \times \tilde{A}_p$ denotes a fuzzy relation in $X_1 \times X_2 \times \dots \times X_p$. The membership function of this fuzzy relation can be characterized as

$$\mu_{\tilde{A}_1 \times \tilde{A}_2 \times \dots \times \tilde{A}_p}(x_1, x_2, \dots, x_p) = \mu_{\tilde{A}_1}(x_1) * \mu_{\tilde{A}_2}(x_2) * \dots * \mu_{\tilde{A}_p}(x_p) \tag{1}$$

where “ * “ is one of t–norm operators. In practice, t-norm operator is selected as the fuzzy intersection “min”. Further information related to various t-norm operators can be found in [60],[62].

For a two-dimensional space (p = 2), a fuzzy relation $\tilde{\mathrm{R}}$ on $X \times X$ is a fuzzy set on the Cartesian product $X \times X$ where $X = [x_1, x_2, \dots, x_m]$ corresponds to a sample data set with n samples and m features. Here, each data sample (or observation) can be showed as a row vector with

$$x_i = \{x_{i1}, x_{i2}, \dots, x_{im}\} \quad \text{i = 1, 2, …, n} \tag{2}$$

Each data sample is an *m*-dimensional vector that has mostly different units, so each feature vector should be normalized or standardized to a unified scale before the classification procedure [35].

Mathematically, a relation matrix $\tilde{R}$ on $X \times X$ can be constituted by various methods such as cosine amplitude, max-min, exponential similarity coefficient, correlation coefficient, scalar product, and nonparametric approaches [59,60,61,62,63,64]:

$$r_{ij} = \tilde{R}(x_i, y_j) \qquad \text{i, j = 1, 2, …., n} \tag{3}$$

In this study, to assign membership values in $\tilde{R}$, a new approach based Hausdorff metric is proposed as well as cosine amplitude, max-min and correlation coefficient.

For a correlation coefficient approach as similar to “*Correlation Coefficient”* in statistics, a fuzzy relation from X to X is evaluated by

$$r_{ij} = \frac{\sum_{k=1}^{n} |x_{ik} - \bar{x}_\iota| |x_{jk} - \bar{x}_j|}{\sqrt{\sum_{k=1}^{n} (x_{ik} - \bar{x}_i)^2} \sqrt{\sum_{k=1}^{n} (x_{jk} - \bar{x}_j)^2}} \qquad \text{i, j = 1, 2, …, m} \tag{4}$$

where $\bar{x}_i$ and $\bar{x}_j$ are averages of features i^{th} and j^{th}, x_{ik} and x_{jk} are k^{th} observations of features i^{th} and j^{th}, respectively.

By means of cosine-amplitude, $\tilde{R}$ can be defined as

$$r_{ij} = \frac{|\sum_{k=1}^{n} x_{ik} x_{jk}|}{\sqrt{(\sum_{k=1}^{n} x_{ik}^2)(\sum_{k=1}^{n} x_{jk}^2)}} \qquad \text{i, j = 1, 2, …, m} \tag{5}$$

By using max and min operators, $\tilde{R}$ is constructed as

$$r_{ij} = \frac{\sum_{k=1}^{n} \min(x_{ik}, x_{jk})}{\sum_{k=1}^{n} \max(x_{ik}, x_{jk})} \qquad \text{i, j = 1, 2, …, m} \tag{6}$$

In terms of being alternative to the other metrics, $\tilde{R}$ can be constituted by Hausdorff distance ($\mathrm{d}_{\mathrm{ij}}^{\mathrm{H}}$) between features x_{i} and x_{j} [65]. From the characteristic of Hausdorff metric, the distances $\mathrm{d}_{\mathrm{i,j}}$ and $\mathrm{d}_{\mathrm{j,i}}$ between x_{i} and x_{j} might not be equal each other. Therefore, the relative distance is to evaluate as the average of $\mathrm{d}_{\mathrm{i,j}}$ and $\mathrm{d}_{\mathrm{j,i}}$, $\mathrm{d}_{\mathrm{ij}}^{\mathrm{H}} = \mathrm{d}_{\mathrm{i,j}}+\mathrm{d}_{\mathrm{ji}}/2$. Let $\mathrm{d}_{\min}$ and $\mathrm{d}_{\max}$ be minimum and maximum distances among observations of x_{i} and x_{j} ,respectively; then the membership values of $\tilde{R}$ can be evaluated as follow [38]:

$$\mathrm{r}_{\mathrm{ij}} = 1 - \frac{\mathrm{d}_{\mathrm{ij}}^{\mathrm{H}} - \mathrm{d}_{\min}}{\mathrm{d}_{\max} - \mathrm{d}_{\min}} \tag{7}$$

3.1.2. Composition

In order to get the certain number of classes on the fuzzy relations, the composition operations are required. Similar to the classical relations, the composition operations can be applied to the fuzzy relations too. For instance, let $X \times Y$ and $Y \times Z$ be the fuzzy relations on $\tilde{R}$ and $\tilde{S}$, respectively. Regarding to $\tilde{R}$ and $\tilde{S}$, any $\tilde{T} = \tilde{R} \circ \tilde{S}$ are constituted by means of certain composition operations on $X \times Z$. To do this, two well-known composition operations, max – min and max – product, can be used. The other composition operators can be found in [59,60, 61,62,63].

Generally, the fuzzy max – min and max-product compositions are defined by set-theoretic and membership notations as follows:

$$\mu_{\tilde{T}}(x,z) = \vee_{y \in Y}(\mu_{\tilde{R}}(x,y) \wedge \mu_{\tilde{S}}(y,z)) \text{ or } \mu_{\tilde{T}}(x,z) = \max - min_{y \in Y}(\mu_{\tilde{R}}(x,y), \mu_{\tilde{S}}(y,z)) \tag{8}$$

$$\mu_{\tilde{T}}(x,z) = \vee_{y \in Y}(\mu_{\tilde{R}}(x,y) * \mu_{\tilde{S}}(y,z)) \quad \text{or} \quad \mu_{\tilde{T}}(x,z) = max_{y \in Y}(\mu_{\tilde{R}}(x,y) * \mu_{\tilde{S}}(y,z)) \tag{9}$$

From the above equations, it is noted that if we let $\tilde{S} = \tilde{R}$, then $\tilde{T} = \tilde{R} \circ \tilde{R}$ will be defined on X×X [38]. In this study, to get more significant classes during the reduction of feature matrix, both composition operations are utilized.

3.1.3 Fuzzy Tolerance and Equivalence Relations

Let $\tilde{R}$ be a fuzzy relation defined on a single universe X. In this case, there are some important algebraic properties: *Reflexivity*, *Symmetry* and *Transitivity*. These properties can be showed as matrix relations as follows [38]:

Reflexivity: For $\forall x_i \in X, (x_i, x_i) \epsilon \tilde{R}$ (10)

or $\mu_{\tilde{R}}(x_i, x_i) = 1$ (10.a)

Symmetry: For $\forall (x_i, x_j) \epsilon \tilde{R}$ and $(x_j, x_i) \epsilon \tilde{R}$ (11)

or $\mu_{\tilde{R}}(x_i, x_j) = \mu_{\tilde{R}}(x_j, x_i)$ (11.a)

Transitivity: For $\forall (x_i, x_j), (x_j, x_k) \epsilon \tilde{R}$,

$\tilde{R}(x_i, x_j) \wedge \tilde{R}(x_j, x_k) \leq \tilde{R}(x_i, x_k)$ (12)

or $\mu_{\tilde{R}}(x_i, x_j) = \lambda_1$ and $\mu_{\tilde{R}}(x_j, x_k) = \lambda_2 \Rightarrow \mu_{\tilde{R}}(x_i, x_k) = \lambda$ (12.a)

where $\lambda \geq \min[\lambda_1, \lambda_2]$.

It can be shown that any *fuzzy tolerance relation*, $\tilde{R}_t$ (also called a proximity relation) on a universe X is a relation that exhibits only the properties of reflexivity and symmetry. $\tilde{R}_t$, can be reformed into an equivalence relation $\tilde{R}_e$ (or similarity) by at most *(m-1)* compositions with itself, where *m* is the size of square matrix $\tilde{R}_t$. That is;

$$\tilde{R}_t^{m-1} = \tilde{R}_t \circ \tilde{R}_t \circ \ldots \circ \tilde{R}_t = \tilde{R}_e \tag{13}$$

Unlike the fuzzy tolerance relation, *an* equivalence relation has also transitivity properties as well as reflexivity and symmetry. Besides, it can be proven that a fuzzy equivalence relation $\tilde{R}_e$, has equivalence classes on $\tilde{R}$. For any given λ-cut level ($\lambda \in [0,1]$) and $r_{ij} = \tilde{R}(x_i, x_j) \geq \lambda$ (i, j = 1, 2, …, m), the equivalence classes of $x_i \in X$ is determined as

$$\tilde{R}\ [x_i]_\lambda = \{x_i | \tilde{R}[x_i] \geq \lambda\} = \{x_i | x_i \tilde{R}_\lambda x_j\} \tag{14}$$

where $x_i \tilde{R}_\lambda x_j$ expresses that x_j (j = 1, 2, …,m) drops into the same class with x_i with respect to $\tilde{R}_\lambda$. Thus, for any given $\lambda \in [0,1]$, the equivalence classes of $\tilde{R}$ give a partition of X called as λ-partition of X.

Consequently, let $\tilde{R}_\lambda = \{(x_i, x_j) | \mu_{\tilde{R}}(x_i, x_j) \geq \lambda\}$ define the λ-cut relation of $\widetilde{R}$, then the columns (features) situated in the same class will be determined with respect to the λ-cut level. That is, if $\mu_{\tilde{R}}(x_i, x_j) \geq \lambda$, then the value on the i^{th} row and j^{th} column of $\tilde{R}_e$ is replaced with 1; otherwise 0. After this assignment called defuzzification, the new matrix consists of only two numbers: 0 and 1. Thus, if all of the values on different columns of defuzzified matrix are equal to each other, they are assigned into same class.

4. ARCHITECTURE OF ANNS CLASSIFIERS BASED THE CROSS-ENTROPY MEASURE

In this study, to estimate the credit scoring model, the feed-forward ANNs with three layers are used. As seen in Fig.1, the first layer receives the feature vectors as inputs. The hidden layer includes the p number of neurons. The last layer gives output $\mathrm{y_c}$ having totally k number of classes $\mathrm{y_c} = [\mathrm{o_1}, \mathrm{o_2}, \ldots, \mathrm{o_k}]$ where c shows the indices of classes ($\mathrm{c} = 1,2,\ldots,\mathrm{k}$). Generally, output $\mathrm{y_c}$ belongs to any one of k classes:

$$\mathrm{y_c} \epsilon [\underbrace{\{1,0,0,,\ldots,0\}}_{\text{Class 1}}, \underbrace{\{0,1,0,\ldots,0\}}_{\text{Class 2}}, \underbrace{\{0,0,1,\ldots,0\}}_{}, \ldots, \underbrace{\{0,0,0,\ldots,1\}}_{\text{Class k}}] \tag{15}$$

Thus, any component $\mathrm{o_c}$ is evaluated as follow:

$$\mathrm{o_c} = \mathrm{f}(\mathrm{W^I}, \mathrm{W^{II}}, \mathrm{x}) = \frac{\mathrm{e}^{\left[\mathrm{b_c^{II}} + \mathrm{w_c^{II}} \mathrm{A}\left(\mathrm{W^I x} + \mathrm{b^I}\right)\right]}}{\sum_{\mathrm{j}=1}^{\mathrm{k}} \mathrm{e}^{\left[\mathrm{b_j^{II}} + \mathrm{w_j^{II}} \mathrm{A}(\mathrm{W^I x} + \mathrm{b^I})\right]}} \in [0,1] \qquad \mathrm{c} = 1,2,\ldots,\mathrm{k} \tag{16}$$

For a specific case, k = 2, $\mathrm{y_c}$ consists of only 0's or 1's (binary values). Essentially, Eq. (16) is the soft-max function (known as a logistic function for k = 2) where

$\mathrm{W^I}$: A matrix consists of all the weights among inputs and neurons in the hidden layer.
$\mathrm{W^{II}}$: A matrix consists of all the weight values among the neurons in hidden and output layers.
x: A vector consists of the values of all the features in any state.
$\mathrm{b^I}$: A vector consists of all the bias values for the activation functions in the hidden layer, $\mathrm{b^I} = [\mathrm{b_1^I}\ \mathrm{b_2^I} \ldots \mathrm{b_s^I}]$. Here, as an activation function, the tangent hyperbolic is used.
$\mathrm{b^{II}}$: A vector consists of all the bias values in the soft-max function in the output layer, $\mathrm{b^{II}} = [\mathrm{b_1^{II}}\ \mathrm{b_2^{II}} \ldots \mathrm{b_k^{II}}]$ [38].

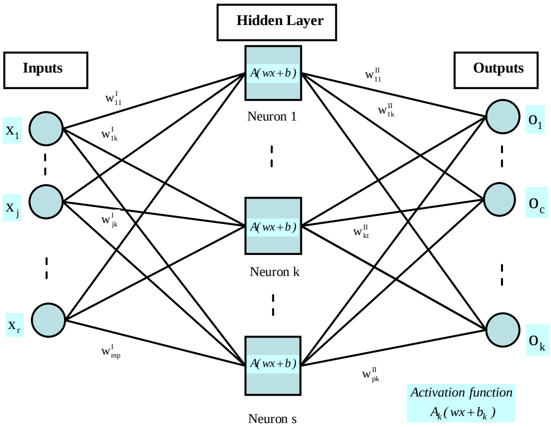

Figure 1. The structure of a feed-forward ANN

According to the above definitions, W^I is structured as follow

$$\mathrm{W^I} = [\mathrm{w^I_1} \;\; \mathrm{w^I_2} \; \ldots \mathrm{w^I_i} \; \ldots \; \mathrm{w^I_s}]' \qquad \text{(s: the number of neurons)} \tag{17}$$

or

$$\mathrm{W^I} = \begin{pmatrix} \mathrm{w}_{1,1} & \mathrm{w}_{1,2} & \cdots & \mathrm{w}_{1,r} \\ \vdots & & \ddots & \vdots \\ \mathrm{w}_{s,1} & \mathrm{w}_{s,2} & \cdots & \mathrm{w}_{s,r} \end{pmatrix} \tag{18}$$

where w_i is a vector, defined as $\mathrm{w^I_i} = \left[\mathrm{w_{i,1}} \;\; \mathrm{w_{i,2}} \; \ldots \; \mathrm{w_{i,r}}\right]$ (i = 1, 2, ..., s). Apparently, w_i includes all the weights between the neuron i^{th} and all the inputs. Similarly, W^{II} is defined as follow:

$$\mathrm{W^{II}} = [\mathrm{w^{II}_1} \;\; \mathrm{w^{II}_2} \; \ldots \mathrm{w^{II}_c} \; \ldots \; \mathrm{w^{II}_k}]' \qquad \text{(k: total number of classes)} \tag{19}$$

or

$$\mathrm{W^{II}} = \begin{pmatrix} \mathrm{w}_{1,1} & \mathrm{w}_{1,2} & \cdots & \mathrm{w}_{1,s} \\ \vdots & & \ddots & \vdots \\ \mathrm{w}_{k,1} & \mathrm{w}_{k,2} & \cdots & \mathrm{w}_{k,s} \end{pmatrix} \tag{20}$$

Eq. (16), $\mathrm{A(w^I x + \; b^I): R^s \rightarrow \; R^s}$ represents a vector function that includes the outputs of all the activation functions as follow [1].

$$\mathrm{A(W^I x + b^I) = G\left(A_1\left(w^I_1 x + b^I_1\right), A_2\left(w^I_2 x + b^I_2\right), \ldots\; , A_s(w^I_s x + b^I_s)\right)} \tag{21}$$

In this framework, $\mathrm{A_j}$ (j=1,2,…, s) is defined as the j^{th} tangent hyperbolic function in the hidden layer:

$$\mathrm{A_j} = \frac{\mathrm{e^{NET_j} + e^{-NET_j}}}{\mathrm{e^{NET_j} - e^{-NET_j}}}\,;\ \mathrm{NET_j = w_j x + b_j} \qquad \mathrm{j} = 1, 2, \ldots, \mathrm{s}. \tag{22}$$

In order to assign the output $\mathrm{y_c = [o_1,\ o_2, \ldots,\ o_k]}$ to only one of k classes, the components of y_c are transformed into the binary numbers {0, 1} as follow:

$$\chi_{\mathrm{o_c}} = \begin{cases} 1, \ \text{if } \mathrm{o_c} = \max\ [\mathrm{o_1},\ \mathrm{o_2}, \ldots,\ \mathrm{o_k}], \\ 0, \ \text{Otherwise.} \end{cases} \tag{23}$$

From Eq. (16), the soft-max function of any component can be considered as the posterior probability of a certain class given any component o_c:

$$P(C_t/o_c) = \frac{e^{\left[b_c^{II}+w_c^{II}A\left(W^I x+b^I\right)\right]}}{\sum_{j=1}^{k} e^{\left[b_j^{II}+w_j^{II}A\left(W^I x+b^I\right)\right]}} \in [0,1] \qquad c,t = 1,2,\dots,k \tag{24}$$

where C_t is one of the predetermined classes. Hence, maximizing the posterior probability in Eq. (24) is equivalent to minimizing the following cross-entropy measure:

$$E(W^I, W^{II}) = -\sum_{i=1}^{N}\sum_{t=1}^{k} y_{i,c}(t)\log P(C_t/o_c) \qquad c,t = 1,2,\dots,k \tag{25}$$

5. APPLICATION

5.1. Data sets and setup of proposed procedure

In the application, to compare the proposed approach with traditional procedure of ANNs, the real-world Australian [66] and German [67] credit data sets (UCI, Machine Learning Repository) are considered. Australian and German benchmark data sets consists of one dependent, fourteen and twenty independent variables regarding to 690 and 1000 loan applicants, respectively. Here, dependent variable expresses whether the credit application is credit-worthy or not, so it is defined as binary number {0, 1} [1]. Also, independent variable set includes totally fourteen categorical and numerical variables for Australian's benchmark data set and totally twenty categorical and numerical variables for German's benchmark data set with different scales. In order to overcome different scale problem, these variables were normalized before training process. During analysis, to control over-fitting and complexity, the credit scoring data was portioned into training, validation and test data as well. All the analysis is given in the different sections and their performances are discussed in detail below. The software of algorithms used in training process is written in MATLAB R2015b.

As is well-known, the excessive number of features might cause the complexity problem, so the eliminating some of them will provide estimating more robust models as well as reducing unnecessary memory allocation in the context of the computer science. Essentially, the memory demand grows exponentially depending on the number of features. In the proposed procedure, to reduce the dimension of feature matrix, the fuzzy relation methodology is used to define the classes of features. In this methodology, the fuzzy relations are constructed by means of some operators such as Hausdorff metric, cosine amplitude, max-min and correlation coefficient. After constructing the fuzzy relations, to get more significant classes both composition operations are utilized. According to the analysis results, the max-min operations provide more efficient classes in the context of feature reduction process. Therefore, the equivalence matrices are constructed by means of max-min operations. In order to determine a specific component of each class, the class average of features is evaluated. These components which are evaluated from each classes are considered as the inputs of ANNs. In analysis, ANNs based cross-entropy and MSE are trained by gradient based algorithms. To control the model complexity during the training process, various techniques are considered such as the cross-validation, stopping and information criteria. After the training, the last step is to determine the best models that present the true classification ratios, information criteria, cross-entropy and MSE. Basic scheme of the proposed procedure is given in Fig. 2.

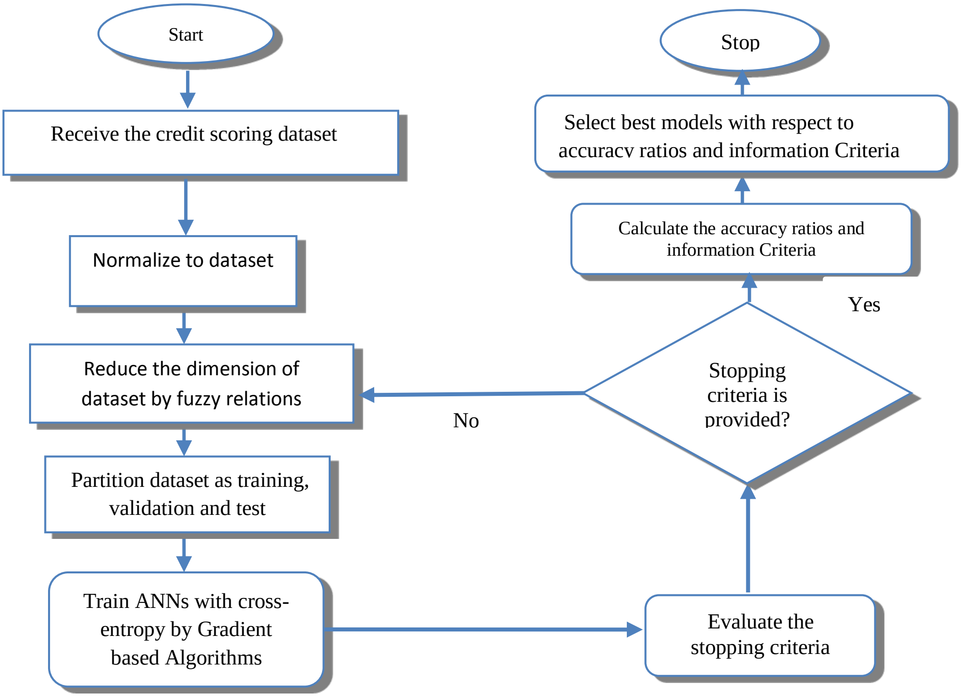
Start
Receive the credit scoring dataset
Normalize to dataset
Reduce the dimension of dataset by fuzzy relations
Partition dataset as training, validation and test
Train ANNs with cross-entropy by Gradient based Algorithms
Evaluate the stopping criteria
Stopping criteria is provided?
No
Yes
Calculate the accuracy ratios and information Criteria
Select best models with respect to accuracv ratios and information Criteria
Stop

Figure 2. The flowchart of proposed procedure

5.2. Analysis

In analysis, to improve the classification performance with respect to Cross-Entropy and MSE, totally five optimization algorithms were used: GD, GDwM, SCG, BFGS and LM. These algorithms were treated by different tuning parameters as well as different number of neurons and layers in the ANNs. More detailed information about tuning parameters can be found in [34],[35],[57],[58]. To control over-fitting and model complexity, early stopping approach with cross-validation was performed as well as considering information criteria. To do that, the data raw data sets were separated into three subsets: training, validation and test. To determine the most efficient number of neurons in the hidden layers, three information criteria AIC, AIC_c and BIC were handled.

All the analysis results are given in Table 1-Table 4 and Table 5 - Table 8 for Australian and German benchmark data sets, respectively. These Tables shows the best model configurations with respect to the fuzzy metric, the partition ratios of training data, the accuracy ratios, the fuzzy metrics, the most efficient number of neurons and inputs. In the Tables, the best performances of models are showed with bold font with respect to information criteria, MSE and accuracy ratios over training and test data sets.

Table 1. Australian's Benchmark Data Set's Performance based MSE

Algorithm	Neuron	Train MSE	Test MSE	AIC	AICc	BIC	Training classes %	Test classes %	Total classes %
GD	**3**	**0.0893**	**0.1560**	**-2568.59**	**-2563.74**	**-2274.26**	**88.8**	**85.5**	**88.1**
GDwM	2	0.0849	0.1451	-1293.22	-1288.60	-1117.93	89.5	80.8	88.3
BFGS	10	0.0907	0.1265	-1002.97	-867.21	-147.49	89.1	85.5	88.4
SCG	2	0.1441	0.1816	-2087.40	-2035.53	-1746.05	87.7	84.1	87.2
LM	2	0.0950	0.1437	-1230.83	-1226.22	-1055.54	87.8	81.7	87.0

Table 2. Australian's Benchmark Data Set's Performance based Cross-Entropy

Algorithm	Neurons	Cross Entropy	Training data MSE	Test data MSE	AIC	AICc	BIC	Training classes %	Test classes %	Total classes %
GD	2	0.2179	0.0886	0.1252	-2609.38	-2607.15	-2411.16	88.2	84.1	87.5
GDwM	**6**	**0.2946**	**0.0874**	**0.1196**	**-1151.50**	**-1108.67**	**-636.09**	**88.2**	**87.0**	**87.7**
BFGS	10	0.3073	0.0899	0.1234	-1007.94	-872.17	-152.46	88.4	87.0	87.8
SCG	2	0.2444	0.0749	0.1286	-1197.72	-1154.79	-682.48	89.5	81.7	88.6
LM	2	-	0.0852	0.1245	-2647.36	-2645.13	-2449.20	89.8	83.7	89.1

Table 3. Australian's Benchmark Data Set's Performance based MSE with feature selection

Algorithm	Fuzzy Metric/ Features	Neuron	Training data MSE	Test data MSE	AIC	AICc	BIC	Training classes %	Test classes %	Total classes %
GD	Cos_Amp/9	2	0.0991	0.1376	-1130.17	-1106.90	-742.28	86.8	81.2	86.1
GDwM	Cos_Amp/9	6	0.0898	0.1217	-1172.70	-1145.18	-752.93	88.9	85.5	88.4
BFGS	**Cos_Amp/9**	**10**	**0.0848**	**0.1161**	**-1100.36**	**-1016.56**	**-404.28**	**89.1**	**87.0**	**88.4**
SCG	Cos_Amp/9	2	0.0744	0.1204	-1220.28	-1217.60	-636.09	86.4	81.2	85.7
LM	Housdorff/8	2	0.0254	0.1395	-1758.87	-1671.90	-1052.42	97.6	84.6	95.7

Table 4. Australian's Benchmark Data Set's Performance based Cross-Entropy with feature selection

Algorithm	Fuzzy Metric/ Features	Neuron	Cross Entropy	Trainin g data MSE	Test data MSE	AIC	AICc	BIC	Trainin g classes %	Test classes %	Total classes %
GD	**Housdorff/9**	**2**	**0.2242**	**0.0899**	**0.1234**	**-1007.94**	**-1872.17**	**-652.46**	**88.4**	**87.5**	**87.8**
GDwM	Housdorff/9	6	0.2690	0.0837	0.1145	-1223.51	-1200.24	-835.62	88.4	82.6	88.0
BFGS	Housdorff/9	10	0.2869	0.0897	0.0996	-1088.73	-1018.78	-445.79	87.0	84.1	86.8
SCG	Housdorff/9	6	0.2595	0.0874	0.1196	-1151.50	-1108.67	-636.09	88.2	87.0	87.7
LM	Housdorff/9	2	-	0.0430	0.1241	-1444.11	-1337.86	-673.90	94.7	84.6	93.3

Analysis results for Australian data set are given between Tables 1 and Tables 4. All the algorithms were worked with 80-10-10 partition ratios, because these ratios provided the best performance for ANNs. According to analysis results, it can be said that the cross-entropy measure improves the accuracy ratios better than MSE. From Table 1 and Table 2, it can be seen that GD and GDwM give the best performance for ANNs, respectively. For Table 3 and Table 4, it can be said that the fuzzy relations help to estimate more robust models with respect to information criteria and accuracy ratios by means of reducing dimension of data set. In Table 3, BFGS gives the best classification result with respect to training data set reduced by cosine amplitude metric. For Table 4, GD brings out the best classification result with respect to data set reduced by Hausdorff metric.

Table 5. German's Benchmark Data Set's Performance based MSE

Algorithm	Neuron	Train MSE	Test MSE	AIC	AICc	BIC	Training classes %	Test classes %	Total classes %
GD	5	0.1491	0.1611	-1300.58	-1263.74	-669.59	79.6	79.0	79.1
GDwM	15	0.1270	0.1745	-989.04	-515.57	-892.56	82.8	78.0	81.2
BFGS	**5**	**0.1665**	**0.1684**	**-1212.21**	**-1175.37**	**-881.22**	**74.9**	**82.0**	**75.3**
SCG	10	0.1442	0.1631	-1107.32	-935.72	-548.98	81.0	78.0	80.3
LM	20	0.1439	0.1621	-669.08	-427.87	746.32	83.8	74.7	81.8

Table 6. German's Benchmark Data Set's Performance based Cross-Entropy

Algorithm	Neurons	Cross Entropy	Training data MSE	Test data MSE	AIC	AICc	BIC	Training classes %	Test classes %	Total classes %
GD	10	0.4467	0.1456	0.1475	-1099.76	-928.16	-156.54	79.8	78.0	78.6
GDwM	5	0.4616	0.1495	0.1777	-1298.21	-1261.37	-667.22	79.6	77.0	78.6
BFGS	20	0.4307	0.1383	0.1680	-700.65	-396.31	-906.27	80.5	77.0	79.4
SCG	**10**	**0.4613**	**0.1493**	**0.1594**	**-1079.49**	**-907.89**	**-1176.81**	**79.3**	**83.0**	**78.9**
LM	20	-	0.1284	0.1698	-979.99	-506.52	-901.62	83.9	75.3	81.9

Table 7. German's Benchmark Data Set's Performance based MSE with feature selection

Algorithm	Fuzzy Metric/ Features	Neuron	Training data MSE	Test data MSE	AIC	AICc	BIC	Training classes %	Test classes %	Total classes %
GD	Max_Min/11	20	0.2053	0.2009	-984.74	-922.92	-183.21	70.8	72.0	70.8
GDwM	**Housdorff/8**	**15**	**0.1659**	**0.1787**	**-1135.12**	**-1063.23**	**-276.74**	**76.0**	**76.0**	**75.6**
BFGS	Max_Min/9	10	0.1880	0.1960	-635.00	-790.48	-385.26	77.1	71.0	76.1
SCG	R_Cor/14	10	0.1665	0.1782	-1112.41	-1029.50	-197.19	76.1	73.0	76.1
LM	Housdorff/8	15	0.1502	0.1805	-376.01	-878.51	-424.04	79.2	73.5	77.8

Table 8. German's Benchmark Data Set's Performance based Cross-Entropy with feature selection

Algorithm	Fuzzy Metric/ Features	Neuron	Cross Entropy	Training data MSE	Test data MSE	AIC	AICc	BIC	Training classes %	Test classes %	Total classes %
GD	R_Cor/14	20	0.4837	0.1585	0.1894	-831.37	-395.37	-993.31	77.3	69.0	76.2
GDwM	R_Cor/14	15	0.5499	0.1853	0.1950	-866.42	-655.26	-503.58	71.1	70.0	71.0
BFGS	Max_Min/11	10	0.4673	0.1525	0.2078	-202.58	-1590.02	-498.11	85.5	72.0	83.3
SCG	**Max_Min/9**	**10**	**0.5004**	**0.1647**	**0.1834**	**-1180.88**	**-1128.24**	**-436.19**	**85.6**	**82.0**	**85.4**
LM	Max_Min/9	15	-	0.1164	0.1743	-658.38	-373.42	-281.72	77,8	69,0	73,8

According to the analysis results given between Tables 5 and Tables 8 for German data set, it can be said that the cross-entropy measure improves the accuracy ratios better than MSE. In addition, ANNs estimated by reduced data set with fuzzy relations provide more robust models with respect to information criteria and accuracy ratios. For German data set, all algorithms were worked with 80-10-10 partition ratios, because it provided the best performance for ANNs. From Table 5 and Table 6, it can be seen that BFGS and SCG give the best performance with respect to accuracy ratios, respectively. For Table 7 and Table 8, it can be said that GDwM and SCG bring out the best results for the training data sets reduced by fuzzy relations based Hausdorff and Max-Min metrics, respectively.

6. RESULTS AND DISCUSSION

In analysis, to control the model complexity in addition to diminishing memory and time consumption, the feature matrix was reduced by the fuzzy relation procedure. To improve the classification performance of ANNs, they were trained by various gradient based algorithms considering two risk functions: cross-entropy and MSE. In the context of generalization and estimating more robust models, the whole data set was portioned into three subsets considering cross-validation methodology: training, validation and test. Training of ANNs was stopped automatically as soon as the error level over validation data grows. Thus, this approach allows estimating more accurate models for ANNs. In analysis, the efficient number of neurons was determined by information criteria that automatically penalize the excessive complex models.

According to analysis results, the gradient based algorithms showed superior performances to each other with respect to different configurations. Especially, ANNs with cross-entropy provides better performances than ANNs with MSE. Besides, the procedure based the fuzzy relation have exposed significant components by reducing the dimension of original feature matrix. Thus, this approach helps to control the complexity of model at the excessive number of inputs. As seen in the above tables, it can be seen that all the algorithms are able to achieve plausible accuracy ratios for training data; however, their efficiencies reduce a little bit for test data. Actually, this reduction is acceptable, because test data can be considered as new individual credit applications, and it is not introduced to ANNs before. In addition, the training procedure utilized validation data set for estimating more general models generalization. From the analysis results, the best configurations of training algorithms are given in Table 9.

Table 9. Comparison of the best configurations for all of benchmark data sets

Benchmark Data Set	Algorithm	Fuzzy Metric/ Features	Neuron	Cross Entropy	Training data MSE	Test data MSE	AIC	AICc	BIC	Training classes %	Test classes %	Total classes %
Australian	GD	Housdorff/ 8	2	0.2242	0.0899	0.1234	-1007.94	-1872.17	-652.46	88.4	87.5	87.8
German	SCG	Max_Min/ 11	10	0.5004	0.1647	0.1834	-1180.88	-1128.24	-436.19	85.6	82.0	85.4

7. CONCLUSIONS

The proposed approach provides substantial advantages in terms of feature selection with fuzzy relations, training ANNs and controlling complexity. Firstly, the fuzzy relations require less memory and time consumption by reducing the dimension of dataset as well as controlling the model complexity. In addition, the fuzzy relations allow the researchers to use different metrics for establishing the similarity between features. Secondly, the information criteria help to determine the efficient number of neurons in the hidden layer. Thus, ANNs based cross-entropy and MSE bring out more reliable and robust models in terms of classification accuracy and model complexity. In addition, the proposed procedure can be easily adapted to various training approaches according to the structure of activation functions and risk functions. As a result, the proposed procedure provides the best model configuration for ANNs in terms of reliability and complexity. In addition, this procedure can be easily applied to another credit data set whose predetermined classes consist of different cases.

In spite of superior advantages, the proposed approach requires an expert knowledge, because it composes of many steps such as preparing data matrix, extracting to features, reducing the dimension of feature matrix, training ANNs, selecting the best model configurations, interpreting and discussing the results. Therefore, an interdisciplinary collaboration is inevitable to construct this kind of expert system thoroughly. In this context, the software of proposed procedure can help users to control this complicated system easily, and reduce time consumption related to adjusting some parameters or functional structures. To validate the robustness of proposed approach, more comprehensive data sets should be handled, but this a challenge due to some legal restrictions. In addition, many benchmark data sets have not enough to explanatory features related to customer information. Hence, these challenges prompt us to make collaboration with experts and financial institutions in terms of making deeply analysis with more comprehensive credit data sets.

As a future direction, we are planning to make collaboration with the financial institutes to reach more comprehensive credit data sets. Also, the studies related to modifying the proposed procedure will be continued in terms for developing more efficient algorithms and estimating more robust models via novel hybrid artificial intelligence techniques. In addition, a user-friendly interface will be designed for developed procedures by MATLAB.

REFERENCES

[1] D. Soydaner, O. Kocadağlı, Artificial neural networks with gradient learning algorithm for credit scoring, Istanbul University Journal of the School of Business, 44(2) (2015) 3-12.

[2] N. C. Hsieh, An integrated data mining and behavioral scoring model for analyzing bank customers. Expert systems with applications, 27(4) (2004) 623-633.

[3] X. Zhu, J. Li, D. Wu, H. Wang, C. Liang, Balancing accuracy, complexity and interpretability in consumer credit decision making: A C-TOPSIS classification approach, Knowledge-Based Systems, 52 (2013) 258-267.

[4] F. L. Chen, F. C. Li, Combination of feature selection approaches with SVM in credit scoring, Expert Systems with Applications, 37(7) (2010) 4902-4909.

[5] G. Wang, J. Hao, J. Ma, H. Jiang, A comparative assessment of ensemble learning for credit scoring, Expert Systems with Applications, 38(1) (2011) 223-230.

[6] K. Bijak, L. C. Thomas, Does segmentation always improve model performance in credit scoring?, Expert Systems with Applications, 39(3) (2012) 2433–2442.

[7] B. Baesens, R. Setiono, C. Mues, J. Vanthienen, Using neural network rule extraction and decision tables for credit-risk evaluation, Management Science, 49(3) (2003) 312-329.

[8] J. R. Quinlan, C4.5: Programs for machine learning, San Francisco, CA, USA, 1993.

[9] M. G. Tsipouras, T. P. Exarchos, D. I. Fotiadis, A methodology for automated fuzzy model generation, Fuzzy Sets and Systems, 159(23) (2008) 3201-3220.

[10] T. Harris, Credit scoring using the clustered support vector machine, Expert Systems with Applications, 42(2) (2015) 741–750.

[11] J. M. Tomczak, M. Zieba, Classification restricted boltzmann machine for comprehensible credit scoring model, Expert System Applications, 42(4) (2015) 1789–1796.

[12] F. C. Tsai, Combining cluster analysis with classifier ensembles to predict financial distress, Information Fusion, 16 (2014) 46-58.

[13] A. B. Hens, M. K. Tiwari, Computational time reduction for credit scoring: An integrated approach based on support vector machine and stratified sampling method, Expert Systems with Applications, 39(8) (2012) 6774–6781.

[14] W. Chen, C. Ma, L. Ma, Mining the customer credit using hybrid support vector machine technique, Expert Systems with Applications, 36(4) , (2009) 7611–7616.

[15] S. T. Lou, B. W. Cheng, C. H. Hsieh, Prediction model building with clustering-launched classification and support vector machines in credit scoring, Expert Systems with Applications, 36(4) (2009) 7562-7566.

[16] C. L. Huang, M. C. Chen, C. J. Wang, Credit scoring with a data mining approach based on support vector machines, Expert Systems with Applications, 33(4) (2007) 847-856.

[17] Z. Huang, H. C. Chen, C. J. Hsu, W. H. Chen, S. S. Wu, Credit rating analysis with support vector machines and neural networks: a market comparative study, Decision Support Systems, 37 (2004) 543-558.

[18] Y. Peng, G. Wang, G. Kou, Y. Shi, An empirical study of classification algorithm evaluation for financial risk prediction, Applied Soft Computing, 11(2) (2011) 2906-2915.

[19] H. Zhu, P. A. Beling, G. A. Overstreet, A Bayesian framework for the combination of classifier outputs, The Journal of the Operational Research Society, 53(7) (2002) 719–727.

[20] A. Bazmara, S. S. Donighi, Bank customer credit scoring by using fuzzy expert system, I.J. intelligent systems and applications, 11 (2014) 29-35.

[21] J. Abellan, J. G. Catellano, A comparative study on base classifiers in ensemble methods for credit scoring, Expert Systems with Applications, 73 (2017) 1-10.

[22] Z. F. Liu, S. Pan, Fuzzy-Rough Instance Selection Combined with Effective Classifiers in Credit Scoring, Neural Processing Letters, 47(1) (2018) 193-202.

[23] M. B. Gorzalczany, F. Rudzinski, A multi-objective genetic optimization for fast, fuzzy rule-based credit classification with balanced accuracy and interpretability, Applied Soft Computing, 40 (2016) 206-220.

[24] J. J. Huang, G. H. Tzeng, C. S. Ong, Two-stage genetic programming (2SGP) for the credit scoring model, Applied Mathematics and Computation, 174(2) (2006) 1039-1053.

[25] K. Crockkett, Z. Bandar, D. Mclean, J. O'Shea, on constructing a fuzzy inference framework using crisp decision trees, Fuzzy Sets and Systems, 157(21) (2006) 2809-2832.

[26] M. Ala'raj, M. F. Abbod, Classifiers consensus system approach for credit scoring, Knowledge-Based Systems, 104 (2016) 89–105.

[27] H. Xiao, Z. Xiao, Y. Wang, Ensemble classification based on supervised clustering for credit scoring. Applied Soft Computing, 43(C) (2016) 73–86.

[28] S. M. Sadatrasoul, M. Gholamian, M. Siami, Z. Hajimohammadi, Credit scoring in banks and financial institutions via data mining techniques: A literature review. Journal of AI and Data Mining, 1(2) (2013) 119–129.

[29] A. Marqués, V. García, J. S. Sánchez, Exploring the behaviour of base classifiers in credit scoring ensembles, Expert Systems with Applications, 39(11) (2012) 10244–10250.

[30] G. Wang, J. Ma, L. Huang, K. Xu, Two credit scoring models based on dual strategy ensemble trees, Knowledge-Based Systems, 26 (2012) 61–68.

[31] C. Hung, J.H. Chen, A selective ensemble based on expected probabilities for bankruptcy prediction. Expert Systems with Applications, 36(3, Part 1) (2009) 5297– 5303.

[32] L. Nanni, A. Lumini, An experimental comparison of ensemble of classifiers for bankruptcy prediction and credit scoring, Expert Systems with Applications, 36(2, Part2) (2009) 3028–3033.

[33] L. Yu, S. Wang, K. K. Lai, Credit risk assessment with a multistage neural network ensemble learning approach, Expert Systems with Applications, 34(2) (2008) 1434-1444.

[34] O. Kocadagli, A Novel Hybrid Learning Algorithm For Full Bayesian Approach of Artificial Neural Networks, Applied Soft Computing, 35 (2015) 1 – 958.

[35] C. Bishop, Neural networks for pattern recognition, Oxford university press, United Kingdom, (2010).

[36] C. Ong, J. Huang, G. Tzeng, Building credit scoring models using genetic programming, Expert systems with applications, 29(1) (2005) 41-47.

[37] J. K. Sengupta, Measuring efficiency by a fuzzy statistical approach, Fuzzy Sets and Systems, 46(1) (1992)73-80.

[38] O. Kocadagli, R. Langari, Classification of EEG signals for epileptic seizures using hybrid artificial neural networks based wavelet transforms and fuzzy relations, Expert Systems with Applications, 88 (2017) 419-434.

[39] K. K. Lai, L. Yu, S. Y. Wang, , L. G. Zhou, Credit risk analysis using a reliability-based neural network ensemble model, Lecture Notes in Computer Science, 4132 (2006) 682-690.

[40] O. Akbilgic, H. Bozdagan, A New Supervised Classification of Credit Approval Data via the Hybridized RBF Neural Network Model Using Information Complexity, Data Science, Learning by Latent Structures and Knowledge Discovery, (2015) 13-27.

[41] A. Blanco, R. Pino-Mejias, J. Lara, S. Rayo, Credit scoring models for the microfinance industry using neural networks: Evidence from Peru, Expert systems with applications, 40 (2013) 356-364.

[42] S. Oreski, D. Oreski, G. Oreski, Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment, Expert systems with applications, 39 (2012) 12605-12617.

[43] L. M. Silva, J. Marques de Sá, L. A. Alexandre Data classification with multilayer perceptrons using a generalized error function, Neural Networks, 21 (2008) 1302 – 1310.

[44] T. S. Lee, I. F. Chen, A two-stage hybrid credit scoring model using artificial neural networks and multivariate adaptive regression splines, Expert systems with applications, 28(4) (2005) 743-752.

[45] R. Malhotra, K. D. Malhotra, Evaluating consumer loans using neural networks, Omega & Science Direct, 31(2) (2003) 83-96.

[46] D. West, Neural network credit scoring models, Computers & operations research, 27, 2000, pp. 1131-1152.

[47] V. S. Desai, J. N. Crook, G. A. J. Overstreet, A comparison of neural networks and linear scoring models in the credit union environment, European journal of operational research, 95 (1996) 24-37.

[48] B. Mirkin, Mathematical classification and clustering. Kluwer Academic Publishers, 1996, pp. 74-76.

[49] R. Saia, S. Carta, An Entropy Based Algorithm for Credit Scoring, Lecture Notes in Business Information Processing, Springer, Cham, 268, 2016, pp. 263-276.

[50] Q. Du, K. Nie, Z. Wang, Application of Entropy-Based Attribute Reduction and an Artificial Neural Network in Medicine: A Case Study of Estimating Medical Care Costs Associated with Myocardial Infarction, Entropy, 16(9) (2014) 4788-4800.

[51] H. Bozdogan, Akaike's information criterion and recent developments in information complexity, Journal of mathematical psychology, 44(1) (2000) 62-91.

[52] S. Geman, E. Bienenstock, R. Doursat, Neural Networks and the Bias/Variance Dilemma, Mass. Inst. Technol. 4 (1), 1992, pp. 1–58.

[53] R. J. May, H. R. Maier, G. C. Dandy, T. M. K. G. Fernando, Non-linear variable selection for artificial neural networks using partial mutual information. Environmental Modeling & Software, 23 (2008) 1312-1326.

[54] M. Qi, G. P. Zhang, An investigation of model selection criteria for neural network time series forecasting, European journal of operational research, 132 (2001) 666-680.

[55] U. Anders, O. Korn, Model selection in neural networks. Neural networks, 12 (1999) 309-323.

[56] N. Murata, S. Yoshizawa, S. Amari, Network information criterion-determining the number of hidden units for an artificial neural network model. IEEE transactions on neural networks, 5(6) (1994) 865-872.

[57] R. M. Golden, Mathematical methods for neural network analysis and design. The MIT press, England, 1996.

[58] M. Moller, A scaled conjugate gradient algorithm for fast supervised learning, Neural networks, 6(4) (1993) 525-533.

[59] A. Blanco, M. Delgado, I. Requena, Identification of fuzzy relational equations by fuzzy neural networks, Fuzzy Sets and Systems, 71(2) (1995) 215-226.

[60] L. X. Wang, A Course in Fuzzy-Systems and Control, Prentice-Hall Inc, Eastbourne, 1997.

[61] P. Liu, The fuzzy associative memory of max-min fuzzy neural network with threshold, Fuzzy Sets and Systems, 107(2) (1999) 147-157.

[62] D. Dubois, P. Henri, Fundamentals of Fuzzy Sets, Kluwer Academic Publishers, Boston, 2000, pp. 233 – 288.

[63] T. J. Ross, Fuzzy Logic with Engineering Applications, McGraw-Hill, Inc., Singapore, 2004.

[64] L. Zadeh, Similarity Relations and Fuzzy Orderings, Information Sciences, 3(2) (1971) 177 – 200.

[65] J. W. Rucklidge, Efficiently Locating Objects Using the Hausdorff Distance, International Journal of Computer Vision, 24(3) (1997) 251-270.

[66] MachineLearningRepository,StatlogData,http://archive.ics.uci.edu/ml/datasets/Statlog+(Australian+Credit+Data).UCI.

[67] MachineLearningRepository,StatlogData,http://archive.ics.uci.edu/ml/datasets/Statlog+(German+Credit+Data). UCI.